THE UNIVERSITY OF ALBERTA

CONTROL SYSTEM DESIGN AND ANALYSIS PROGRAM

by

© RICHARD A. FARWELL

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE

MASTER OF SCIENCE

IN

PROCESS CONTROL (CHEMICAL ENGINEERING)

DEPARTMENT OF CHEMICAL AND PETROLEUM ENGINEERING

EDMONTON, ALBERTA

SPRING, 1970

THE UNIVERSITY OF ALBERTA

CONTROL SYSTEM DESIGN AND ANALYSIS PROGRAM

by

© RICHARD A. FARWELL

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE

MASTER OF SCIENCE

IN

PROCESS CONTROL (CHEMICAL ENGINEERING)

DEPARTMENT OF CHEMICAL AND PETROLEUM ENGINEERING

EDMONTON, ALBERTA

SPRING, 1970

UNIVERSITY OF ALBERTA

FACULTY OF GRADUATE STUDIES

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies for acceptance a thesis entitled "Control System Design and Analysis Program" submitted by Richard A. Farwell in partial fulfilment of the requirements for the degree of Master of Science.

ABSTRACT

A digital computer program to provide the control engineer with an interactive system for the analysis and design of linear continuous feedback control systems has been written in FORTRAN. The program, implemented on an IBM 1800 digital computer operating under the Time Sharing Executive system, required 9700 words of core storage plus disk storage amounting to about 400 sectors.

The control systems design and analysis program (CSDAP) uses the block diagram, transfer function representation technique, familiar to control engineers, for problem formulation and subsequent data entry. The individual transfer functions are entered through the user's keyboard where they are immediately error checked before being included in the system definition data. After the transfer functions have been entered the user must then decide upon the technique or techniques to be used in the design and analysis of the control system and select accordingly. The user can specify one or more of the following options:

a) The open or closed loop transient response of the controlled variable to a change in set point or load variable can be determined. Various performance criteria can be calculated from this response data.

b) The frequency response of the open and closed loop can be calculated and the results can be displayed as a Bode, Nyquist or Log-Modulus (Nichol's) plot. In the

case of the Bode diagram the gain and phase margins and crossover frequencies are included.

c) The root locus values are calculated for both positive and negative values of the gain. The results are available as a root locus diagram or as a listing of the frequency coordinates and gain values.

The calculated values are available as a display on the oscilloscope, as graphs drawn by the Calcomp Digital Plotter or as a tabular listing on the line printer or as any combination of these methods.

The program has been designed with a strong emphasis on an interactive "conversational" mode of operation so a considerable amount of user assistance has been provided as part of the program. This assistance, which is available as a user selected option for all sections of CSDAP, minimizes the prior study required to utilize the program. The program allows the user to alter particular parameters of interest without re-entering the remainder of the parameters which remain unchanged.

A complete Systems Manual for CSDAP is available so modifications to the existing program, for use on an IBM 1800 operating under a different executive system or another machine, should not be difficult.

ACKNOWLEDGEMENTS

The author wishes to express sincere appreciation to Dr. R.K. Wood for his supervision of this work and to the other members of the Department of Chemical and Petroleum Engineering who were consulted.

The author also expresses a special thanks to Mrs. Susan Ison for the typing of all stages of the manuscript.

TABLE OF CONTENTS

Page

LIST OF FIGURES

LIST OF TABLES

CHAPTER I

INTRODUCTION

The conventional use of a digital computer for analysis and design problems suffers from two major disadvantages.

First, the usual form of supplying information to the computer programs is by means of cards or tapes. Those which must be prepared according to rigid formats dictated by the program are often alien to the design engineer. Even in the case of programs in which an effort has been made to simplify the required inputs that are used, the designer is usually plagued by input data errors until he develops a certain familiarity with the particular programs. Errors due to improper input data preparation disrupt the design iteration cycle and frustrate the designer.

The second and more important disadvantage is the delay, due to a long turn-around time, in getting the computational results back to the designer. This delay undermines the effectiveness of the iterative design process and makes it impossible for the user to try out new ideas rapidly.

A means of bypassing these man-computer communication problems lies in the use of on-line computer consoles which have graphical input and/or output capability.

Conversational programs using such equipment allow the user to input information into the system that may take

the form of command functions or data. The solution is then available in the form of a graph or table of printed results. Modifications to the problem parameters based on the results of the previous calculations can be entered and the problem rerun.

It matters little to the user how efficiently such conversational flexibility is achieved, as long as the programs are reasonably responsive and easy to use. From the systems point of view, there will always be a serious concern for the amount of core memory and bulk storage required and the execution speed.

This thesis describes how a set of programs that have been developed provides a versatile means of using a digital computer to analyze and design linear control systems. The programs are written in the FORTRAN IV language and were implemented on an IBM 1800 computer located in the Department of Chemical and Petroleum Engineering.

The IBM 1800 computer operates under the Time Sharing Executive (TSX) operating system which allows a background or nonprocess program to utilize any computing time that is not being used to monitor and control on-line processes. Peripheral devices on the computer such as disk bulk storage, card reader/punch, line printer, keyboard/typewriter, digital plotter, or storage display oscilloscope allow the analysis programs to operate under the TSX monitor in a conversational mode.

The total system is organized in much the same way that a control engineer might approach the problem. First the transfer functions of all the component blocks in the control loop are found and entered into the system. Once all the transfer functions are in storage, a grouping of all the transfer functions is made and used to form the open loop and closed loop transfer functions of the system. Subsequently, the time response, frequency response, and/or root locus plots can be produced at the user's option. These topics are covered in more detail with examples in the chapters that follow.

Although this thesis briefly describes most of the subroutines used, it does not contain extensive program documentation. The Systems Manual that is available can be consulted for the organizational flowcharts, program linkages and program listings. A User's Manual is included in Appendix A.

CHAPTER II

LITERATURE SURVEY

There are a number of methods that may be used to perform an analysis of the behavior of a control system. Although the majority are graphical techniques, they can be calculated analytically with the use of high speed digital computers. The following sections will discuss some programs and analytical methods that have been used previously.

A. Control Systems Analysis Programs

The last decade has seen a number of programs written to handle the laborious task of control systems analysis. Cummins (12) developed a generalized procedure for synthesizing automatic control systems using some of the fundamental concepts. The procedure was programmed for the Ferranti-Mercury and the IBM 7090 computers. Computations are performed after the system is described by a linearized set of equations.

In the early sixties with a great deal of interest focused on the aerospace industry IBM (15) developed a program for its 7090 computer. This control systems analysis program was written primarily to accomplish real-time flight simulation by digital computational techniques. The program provided a convenient method of obtaining a root locus for linear, continuous or sampled-data systems.

Input data can be in the s-transform and/or z-transform notation. Optional features included the conversion of a polynomial in Laplace notation to sampled-data or z-transform notation and the computation of pole-zero loci and gain loci for a specified gain.

Lofkrantz (24) in an investigation of this IBM program found that routines could be easily incorporated to allow the plotting of the Bode and Nyquist stability graphs. A method for calculating the transient response from the z-transform was also implemented. Some modifications were also made to decrease the number of runs required when calculating the closed loop transfer function and its subsequent transient response.

The analysis and design of multiloop, multivariable, linear control systems requires the use of computers. Coffey (8) introduced algorithms for applying frequency response, time response, and root locus methods to such systems with the aid of a digital computer. He describes a digital program AUTO, which uses the gradient optimization scheme to automatically synthesize compensation for multiloop continuous and digital control systems. This compensation is calculated to achieve a desired open-loop frequency response. The second part of his study is devoted to a discussion of simulating complex systems by means of digital, analog, and hybrid computers with emphasis being placed on the fundamental aspects of simulation.

A recent program developed by Agostinis (1) allows the control engineer to use a digital computer to assist in the analysis and design of linear, time-invariant control systems or multivariable systems involving pure time delay elements. The program uses the matrix formulation of the state variable equation to produce the time domain solution for the output variable. The study gives a number of worked examples, a user's manual and a good section on system documentation including detailed flow charts and program listings.

B. Transient Response with Associated Performance Criteria

The Laplace transformation is the most common means of solving the differential equations describing the performance of a control system. In order to find the transient response it is necessary to invert the Laplace transform. Corrington (10) showed that the transient response for a system described by the ratio of two polynomials in the complex frequency domain could be calculated using a simple linear difference equation. Liou (23) formulates the corresponding state equation from the transfer function, derives a recursive formula for the solution, and gives the result for any desired interval of time. Chen and Yates (6) discuss a matrix formula for the inverse Laplace transformation. After the eigenvalues and coefficients are substituted, and some simple matrix operations are performed, the inverse Laplace transformation of the

function being studied is obtained.

The matrix formula by Chen and Yates has been implemented on a digital computer and some of the algorithms and the program listing appears in the book by Chen and Haas (4) and is also given by Chen and Shieh (5). The formula is general and is particularly advantageous for use on high-order transfer functions. Since many of the control systems studied in the process industries are of a moderately low order, the method of partial fractions could be efficiently used to invert the Laplace transforms. This is a familiar techniques (7, 11, 13, 22, 26, 28, 29, 33) and with the help of routines (17) for polynomial manipulations and a Newton-Bairstow (9) approach to calculating the poles of the system, it could be implemented on a digital computer.

The pure time delay or transportation lag as it is called is frequently found in control systems. It cannot be expressed exactly as a ratio of polynomials. Consequently some form of approximation is desirable before the transient response can be calculated. As some authors (11, 26, 29) show it is frequently approximated by a function with a high order real pole far from the origin, or by a rational function of s that includes the first several terms in the power series expansion. This latter approach is referred to as the Pade approximation.

Some compensation is usually necessary to improve the transient or steady state performance. The principle

difficulty is the establishment of a single criterion for performance that can be used for design. Several papers (25, 27, 30) and a number of textbooks (13, 22, 26, 29) have presented and compared several figures of merit for optimizing the system response. Several of these performance criteria have been included in the program.

C. Root Locus Analysis

The graphical method of constructing the root locus of the closed loop poles, when the gain is varied from zero to infinity, has been well developed (14, 31). An analytic approach for obtaining the root locus with positive and negative gain has also been described (3, 32). These studies show how solutions for various systems consisting of rational polynomial functions of s are obtained. A more recent study shows that the analytical method merits utilization because of the flexibility inherent in the manipulation of algebraic expressions. In this study, Krishnan (21) demonstrates that arrays can be developed so algebraic expressions for the root locus and associated gain can be obtained from a knowledge of the open loop transfer function. Breakaway points on the real axis and the intersection with the imaginary axis can also be obtained directly from the arrays.

D. Other References

During the development of the program a number of manuals associated with the use and operation of the IBM 1800 data acquisition and control system were consulted (15-20).

A number of textbooks (2, 11, 13, 22, 26, 29) were utilized for methods and examples of the frequency response analysis techniques.

CHAPTER III

SYSTEM ORGANIZATION

A. Control System Representation

One of the first steps when analyzing a physical system is to derive a mathematical model which describes the characteristics of the system. This representation of the interrelationships between system components allows the control engineer to use the available mathematical techniques to study the behavior of the system.

The analysis of linear feedback control systems is facilitated by the use of transfer functions and block diagrams. A block diagram is usually described as a functional layout of an entire system in which the transfer functions of the components are shown in the blocks and the flow of signals is shown by lines connecting the blocks. Each block is an independent unit in that the connections between the blocks do not affect the transfer function inside it. Although the block diagram is used to provide a basis for an analog computer solution, it is also useful in the formulation of a problem which is to be solved numerically using a digital computer.

The transfer function is obtained by writing the differential equation relating the output to the input of the block diagram, replacing the operator d/dt by the complex variable s, and solving for the ratio of the output to input. It is one of the most general and convenient

means of characterizing the dynamics of a physical system.

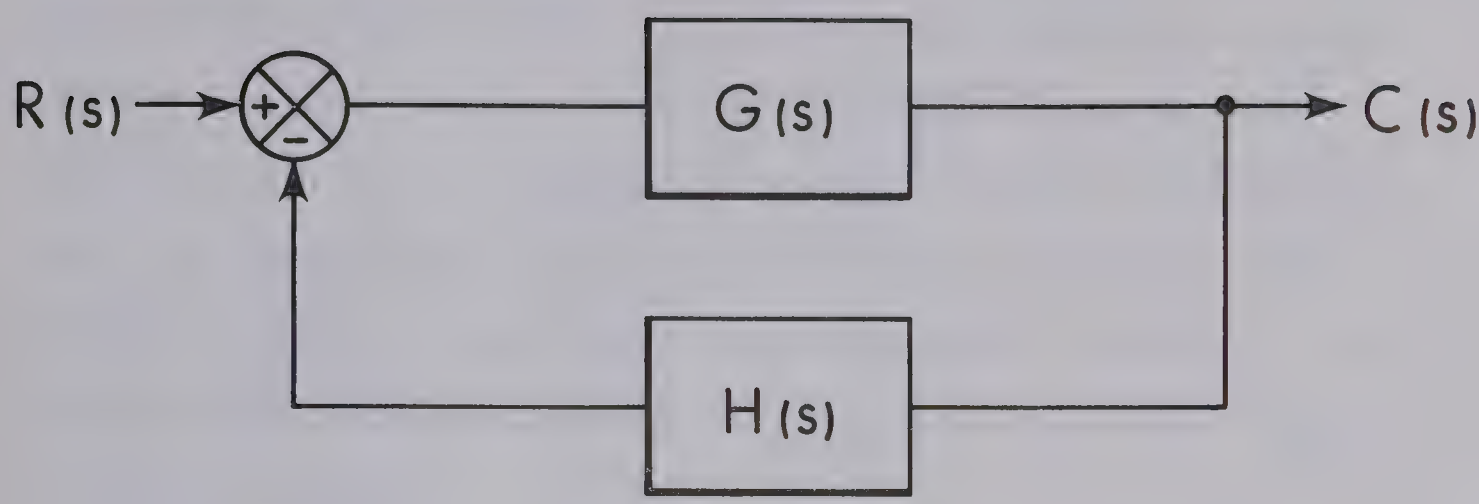
The combination of the block diagram and the transfer function of a physical system provides a pictorial representation of the cause and effect relationship between the input and output of the system.

Although all systems that are characterized by one input and one output may be denoted by a single block connected between the input and output, the advantage of the block diagram concept lies in the fact that feedback control systems are composed of many non-interacting elements whose transfer functions are determined independently. Therefore, an entire system may be represented by the interconnection of the blocks of individual elements, so that their contributions to the overall system performance may be evaluated individually.

In the study of single-loop control systems, certain transfer functions are of special importance. In the general case shown in Figure 3-1, G(s) is referred to as the "forward" transfer function while H(s) is called the "feedback path" transfer function. The relation between the output, C(s), and the input, R(s), of this system can be obtained as shown in Equations 3-1 and 3-2.

$$C(s) = G(s)\,R(s) - G(s)\,H(s)\,C(s) \tag{3-1}$$

$$\frac{C(s)}{R(s)} = \frac{G(s)}{1 + G(s)\,H(s)} \tag{3-2}$$



NOMENCLATURE

R(s)	-	Setpoint or reference input
C(s)	-	Output or controlled variable
G(s)	-	Forward path transfer function
H(s)	-	Feedback path transfer function
G(s) H(s)	-	Open loop transfer function
$\dfrac{G(s)}{1 + G(s)\,H(s)}$	-	Closed loop transfer function
$1 + G(s)\,H(s) = 0$	-	Characteristic equation

FIGURE 3-1 STANDARD FEEDBACK CONTROL SYSTEM BLOCK DIAGRAM

Since the transfer function in Equation 3-2 relates the output to the input with the loop closed, it is referred to as the "closed loop" transfer function of the system. The relationship between the output and the input of the system through the forward path transfer function when feedback action is removed by opening the path between H(s) and the comparator given by the transfer function, G(s) H(s), is known as the "open loop" transfer function. The denominator of Equation 3-2 is referred to as the "characteristic equation".

Seldom is it possible to represent a control system with the simplified block diagram of Figure 3-1. The control engineer is normally interested in analyzing the behavior of a system for disturbances in load variables and the effect of varying control parameters on the overall system performance. In reviewing the complexity of process control systems that might be studied, the author finally decided on the general system representation shown in the block diagram of Figure 3-2.

Once a control system can be represented in the same form as this block diagram, the next step is to determine the "overall" transfer function that relates the output to the setpoint or to either one of the load variables. A symmetric approach for obtaining the overall transfer function for a setpoint change or load change will be described in Section C of this Chapter.

NOMENCLATURE

R(s)	-	Setpoint or reference input
C(s)	-	Output or controlled variable
B(s)	-	Feedback signal
U1(s), U2(s)	-	Load variables or disturbances
C1(s), C2(s)	-	Controller transfer functions
P1(s), P2(s), Ps(s)	-	Process transfer functions
H1(s), H2(s)	-	Feedback element transfer functions
L1(s) L2(s)	-	Load transfer functions

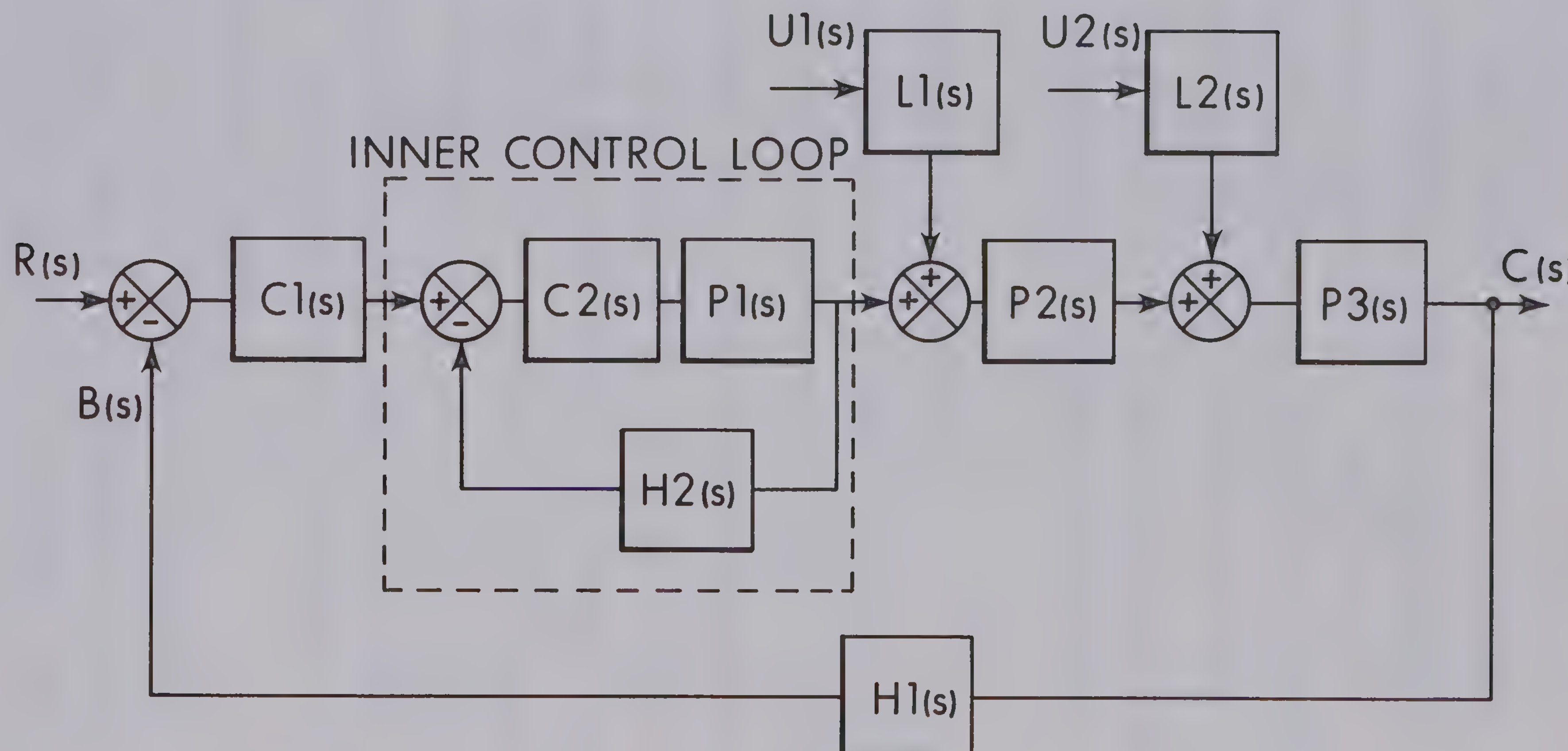


FIGURE 3-2 BLOCK DIAGRAM FOR A MULTILOOP-MULTILOAD SYSTEM

B. Master Program Description

The following three sub-sections will describe the overall system organization of the programs and the manner in which they were implemented. The first section will be a discussion of why a conversational system was used while the other sections will outline how the executive controls the entry into different phases of the total system and how communication between these phases is implemented.

1. Conversational Operating System

Conversational programs were prepared to use terminal input and output facilities for carrying on a dialogue with users. User programs will always require some form of input from the user; either command functions or data. This is an obvious source of error because of lack of experience, typographical mistakes, or transmission errors. It is most important to provide assistance to the user in guiding proper input, reporting detectable errors, and enabling corrections. This later facility is usually reflected by immediate syntax checking.

However, there is one difficulty with providing such input help to a user, namely that depending on the expertise of the user, it may or may not be appreciated. If a user is very adept, excessive verbosity on the part of the program will be considered a severe annoyance and will tend to make the operation slow. If the user is a novice or highly subject to making errors, it is appropriate

to insure proper input.

1.1 Optional User Assistance

By recognizing that a user may optionally need comprehensive input help, a conversational system can satisfy the real-world spectrum of users. This feature has been provided in the present version of the program by means of data switches.

When operating the executive program checks one of the sense switches (determined at system loading time). If this switch is in the "OFF" position, the IBM 1800 console data switches will be used. If the switch is in the "ON" position, the set of digital input switches will be used.

These data switches when set to the "ON" position cause the program to list the possible alternatives or options for the requested function parameter. Table 3-1 lists each section of the analysis program that individual switches control. As the user becomes familiar with various sections of the total program, some of the switches would be left in the "OFF" position.

1.2 Recognition of Input Errors

Error processing is one of the most important features of a conversational program. Those errors or problems occurring during execution which require some decision making or correction can be resolved because the user is available at the terminal. Programs operating in a batch-run environment usually assume that a serious error will

TABLE 3-1

MEANING OF THE DATA SWITCH OPTIONS

Switch Number	Program Section Controlled
1	LIST CONTROL DIGITS AND SWITCH DATA
2	LIST TRANSFER FUNCTION DATA AFTER INPUT
3	INPUT-OUTPUT VARIABLE AND FORCING FUNCTION SELECTION
4	LIST PARTIAL FRACTION DATA
5	REPLOT, AND PLOT OPTIONS FOR TIME RESPONSE
6	LIST TIME RESPONSE DATA ON THE PRINTER
7	PERFORMANCE DATA OUTPUT DEVICE OPTIONS
8	FREQUENCY RESPONSE DATA OPTIONS
9	BODE PLOT OPTIONS
10	LIST FREQUENCY RESPONSE DATA ON THE PRINTER
11	NYQUIST PLOT OPTIONS
12	LOG-MODULUS PLOT OPTIONS
13	ROOT LOCUS PLOT OPTIONS
14	LIST ROOT LOCUS DATA ON THE PRINTER

cause a job abort because the user is usually not present. Even if the user was present he would not be allowed to intervene because it would hold up the job queue excessively.

In the control systems analysis program being described here the data is error checked immediately after the user completes the input operation. If there is an error, an appropriate message is typed out in red print and the user is requested to re-enter the data.

A typical example would be the case where an error had been made by the user when specifying the end time and subsequently the number of subdivisions for the time axis. The correspondence might have appeared as follows with the user response underlined.

```
          ENTER TOTAL TIME
IK
          ERROR IN INPUT - TRY AGAIN
25
          ENTER NUMBER OF TIME DIVISIONS, MAX. = 500
600
          DATA WAS NOTCONSISTENT, TRY AGAIN
60
```

The computer subroutine first found that the user had entered non-numerical characters to specify the end time and so requested the data again. The number of time divisions after it is entered is first checked to make sure it is a positive integer and then it is compared against the limits and in the above case it was greater than the maximum allowable.

1.3 Convenience in Inputting Data

Although the majority of the programs were written in FORTRAN, the author found it necessary to write a number of subroutines associated with the input of data to the program in IBM 1800 assembly language (19).

The first of these was to enable the user to enter a single numeric digit from the typewriter keyboard. This was necessitated because many of the options or branch decisions could be coded using the numeric digits 0, 1, 2, ..., 9. The conventional method meant that the desired numerical character must be typed followed by the end of field character. This extra character provided for additional delay or errors to occur.

The calling sequence of this subroutine INCHR which is listed and flow charted in the systems manual is given below.

CALL INCHR (IDEN, INMBR, ILOW, IHIGH)

It allows a numerical character INMBR to be entered through the keyboard on logical unit number IDEV. The other two parameters ILOW and IHIGH provide an allowable range for the inputted character. The subroutine reads a single character from the keyboard, checks that it is a numerical character and if it is that it does not lie outside the range provided by the limits. If either of the above errors occurs the routine types a message and requests it again.

Another subroutine FFINP allows the user a less restrictive method data input to his program than is

available in FORTRAN. The subroutine allows integer or real variables to be read in as single values or as a vector of values. The variables are entered separated by a space or comma delimiters. The input field can contain any number of leading and/or trailing blanks but cannot contain any embedded blanks within the variables. Alphanumerical data can also be read into the integer or real variables under the A2 or A4 format respectively. In the case of an alphanumeric data entry the field begins at the first non-blank character and continues until the field is filled or a common delimiter is encountered. A space is not a legal delimiter for this data entry. The subroutine is fully documented in the Systems Manual.

The final subroutine written in assembler language was TWAIT. This routine when it is called, loops internally until the device specified in the calling sequence is not busy. It is required because a keyboard entry is of higher priority than outputting a message and a request for input might be made before the description of the alternatives was listed for the user.

2. Executive Program Organization

A conversational program permits the user to change his mind by stopping the execution or starting the program over again, and even returning along the calculation path to a previous logical breakpoint. Such flexibility supports the dynamic on-line user environment.

The Control Systems Design and Analysis Program has thus been divided into ten logically self-contained sections as shown in the simplified program flowchart Figure 3.3. This program requests the user to enter the CONTROL DIGIT for the next section of the program to be executed (see Table 3-2). The list of these options can be obtained by the user at the start of a run and optionally during the program execution by having data switch 1 in the "ON" position and entering a CONTROL DIGIT of 0.

The CSDAP System is designed so that as each section is completed it returns to the executive. This means the calculational path can be changed at any logical breakpoint which is the successful completion of one of the sections.

For example we will assume that the user has completed entering the necessary transfer functions, calculated the inverse Laplace transformation, and displayed the time response on the storage scope. It is possible to have the same results plotted onto the digital plotter by repeating only the programs associated with CONTROL DIGIT number 3. Because the plot is just a replot of the data previously calculated for the scope display it is only necessary for the user to enter the plot device, plot direction, and the title for this graph.

3. Communication Between the Program Sections

The communication between the programs and subroutines was achieved by using FORTRAN COMMON and disk FILES.

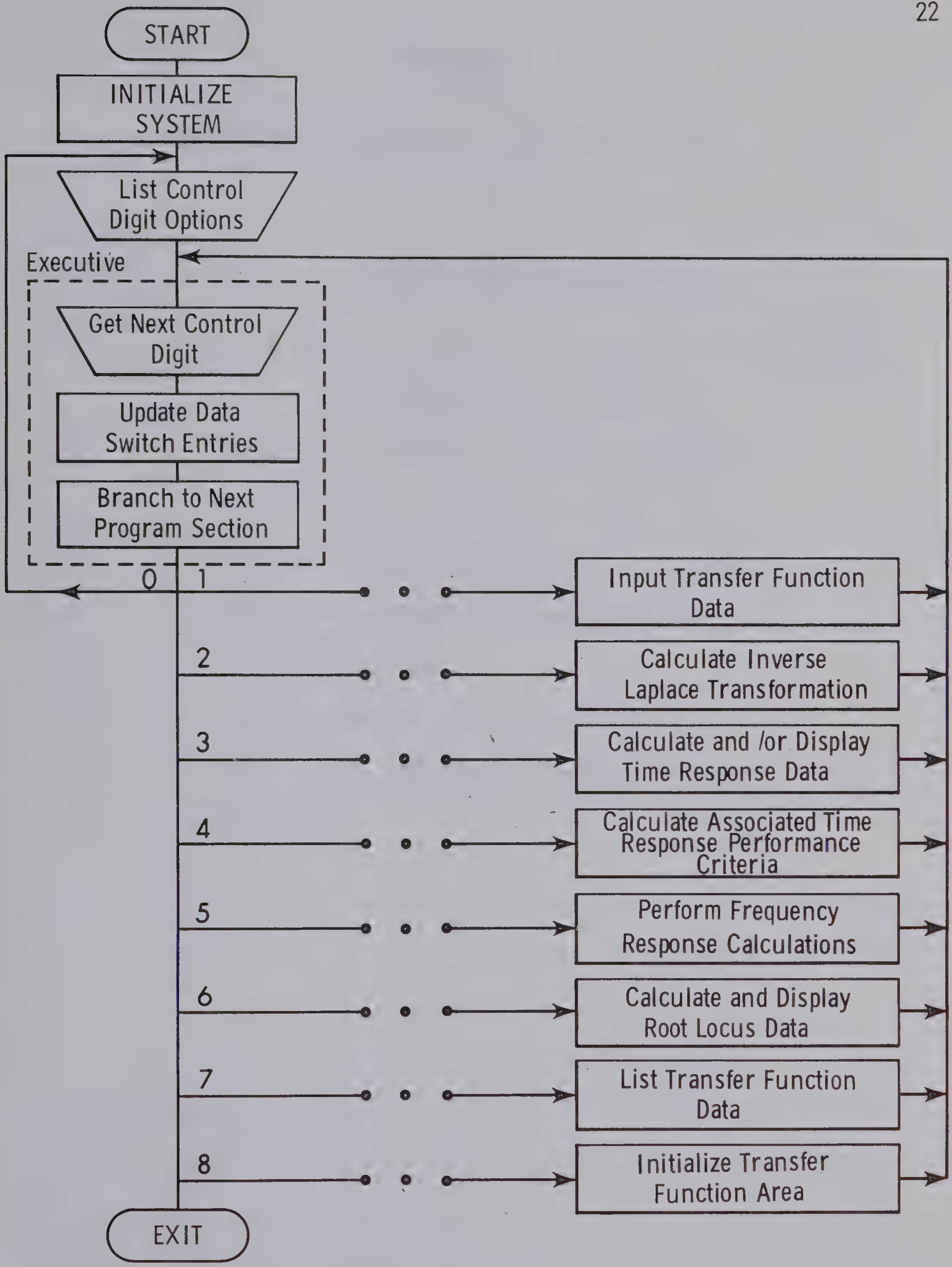


FIGURE 3-3 SIMPLIFIED FLOW DIAGRAM OF THE PROGRAM

TABLE 3-2

LISTING OF CONTROL DIGIT FUNCTIONS

Control Digit	Program Section Entered
0	LIST CONTROL DIGIT AND DATA SWITCH OPTIONS
1	INPUT TRANSFER FUNCTION DATA
2	CALCULATE INVERSE LAPLACE TRANSFORM
3	PLOT THE TIME RESPONSE DATA
4	CALCULATE THE PERFORMANCE CRITERIA
5	FREQUENCY RESPONSE CALCULATIONS
6	ROOT LOCUS PLOT
7	LIST TRANSFER FUNCTION DATA
8	INITIALIZE TRANSFER FUNCTION AREA
9	CALL EXIT

Their use will be briefly described here with full documentation in the Systems Manual.

The COMMON area is defined as four blocks and has therefore kept the COMMON statement in each program as simple as possible. The first block ISTOR is an array of eighty integers that contains variables frequently used by all of the program sections such as the logical unit numbers of the input/output devices, file numbers, array pointers, data switch settings and the names of the transfer functions. It is initialized once at the beginning of a run by the mainline CSDAP for any variables that remain constant during the run such as the logical unit and file numbers. If a variable is required in a program it is available by equivalencing an integer to the correct element of the ISTOR array.

The other three blocks of COMMON are used to store the transfer function data and appear in dimensional form in the program as IC (2, 40), ICONT (200), and DATA (400). Their use will be discussed further in Section C of this Chapter.

A few disk files are used in the system. The main use of these files is for saving the COMMON area and for storing calculated data for replotting the time response, the frequency response diagrams or the root locus diagram. Because the size of core available to the nonprocess user is under ten thousand words, it was found necessary to overlay the common area when not directly working with the

transfer function data. This area was saved and restored into a file on disk by the use of subroutine DATSV.

C. Transfer Functions

1. Block Diagram Transfer Functions

The block diagram that appears in Figure 3.2 has been used as the basis for the transfer function entry into the program. The control system is composed of two control loops and nine blocks with the reference or set-point and two load variables as inputs and one output variable.

To facilitate easy block diagram manipulation, the transfer function of each block is entered as a ratio of polynomial terms in the complex variable s. A general form taken by such a transfer function would be

$$P3(s) = \frac{(a_1 s + a_2)(a_3 s^2 + a_4 s + a_5)(a_6 s^q + a_7 s^{q-1} + \ldots)(\ldots)}{(b_1 s + b_2)(b_3 s^2 + b_a s + b_5)(b_6 s^p + b_7 s^{p-1} + \ldots)(\ldots)} \tag{3-3}$$

The transfer function (Equation 3-3) is identified in the program by the two alphanumeric characters P3 which represents process block number three. A similar designation is used to represent the other individual block transfer functions shown in Figure 3.2.

2. Transportation Delay

The pure time delay or transportation lag as it is sometimes called is frequently found in control systems. It

cannot be expressed exactly as a ratio of polynomials as was the case for the other transfer functions. It is therefore necessary to make some form of approximation which preserves the all pass nature of the pure time delay and gives an essentially linear curve of phase shift versus frequency for the range of frequencies being investigated.

There are several means of achieving the required transfer function for the time delay. One method of approximation involves the replacement of $e^{-\tau s}$ by a rational function of s such that the first terms in the Taylor power series expansion of $e^{\tau s}$ appear in the denominator (Equation 3-4)

$$e^{-\tau s} = \frac{1}{e^{\tau s}} = \frac{1}{1 + \tau s + \tau^2 s^2/2! + \tau^3 s^3/3! + \ldots} \qquad (3\text{-}4)$$

This approximation is reasonably accurate for values of $\omega\tau<1$ but for $\omega\tau>1$ both the magnitude and phase shift differ sharply from the exact representation as shown in the Bode plot (Figure 3-7).

The Padé approximation (26) for the time delay is used to match any required number of coefficients in the power series expansion, by a ratio of zeros and poles. The more accurate approximations require more poles and zeros. A general formula for the Padé approximation appears in Coughanowr and Koppel (11).

The first four orders of the Padé approximation are listed below:

$$e^{-\tau s} = \frac{1 - \tau s/2}{1 + \tau s/2} \quad \text{(First Order)} \tag{3-5}$$

$$e^{-\tau s} = \frac{1 - \tau s/2 + \tau^2 s^2/12}{1 + \tau s/2 + \tau^2 s^2/12} \quad \text{(Second Order)} \tag{3-6}$$

$$e^{-\tau s} = \frac{1 - \tau s/2 + \tau^2 s^2/12 - \tau^3 s^3/120}{1 + \tau s/2 + \tau^2 s^2/12 + \tau^3 s^3/120} \quad \text{(Third Order)} \tag{3-7}$$

$$e^{-\tau s} = \frac{1 - \tau s/2 + \tau^2 s^2/12 - \tau^3 s^3/120 + \tau^4 s^4/1680}{1 + \tau s/2 + \tau^2 s^2/12 + \tau^3 s^3/120 + \tau^4 s^4/1680} \quad \text{(Fourth Order)} \tag{3-8}$$

For these approximations the magnitude of the frequency response agrees with the true magnitude exactly for all τ, and the phase shift is a better approximation over a wider frequency range than the power series approximation (Figure 3-6).

The time delay approximations are handled by a subroutine DLYTM that is entered from the transfer function input section. Instead of a legal transfer function name as shown in the block diagram (Figure 3-2), the letters TD are entered for deadtime. The DLYTM subroutine is then entered and requests the user to enter further information describing the time delay approximation. The time delay can be placed in the forward path-block P3 or in the feedback path-block H1. Once the rational polynomial approximation is calculated it is multiplied by the existing contents of the

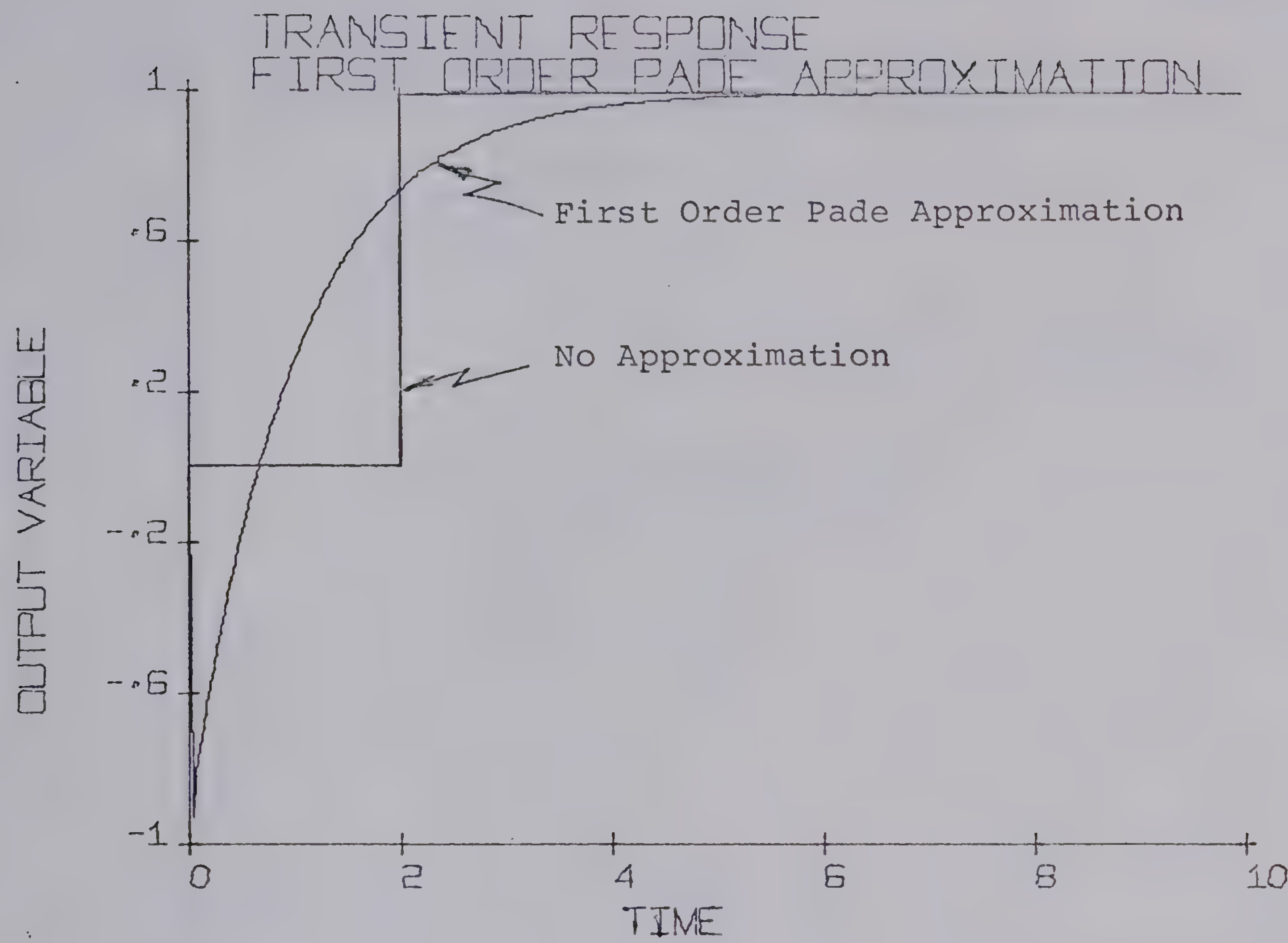
respective block P3 or H1. This makes it possible to have another transfer function already present for this block and dictates that the time delay must be the last transfer function entry for either the P3 or H1 blocks.

The two methods previously discussed for approximating the time delay are both available in the program. For the power series method up to a fourth-degree Taylor series expansion can be used in the denominator. The Padé approximation can take any of the forms described by Equations 3-5 to 3-8 inclusive. Figure 3-4 shows the time response of both the first and second order Padé approximations to a unit step forcing function. This figure compares well with a similar figure given by Buckley (2).

The time response for the fourth-degree power series and Padé approximations is shown in Figure 3-5 where each of the time delay transfer functions had a unit step forcing function applied. The Bode plots of both these approximations is given in Figures 3-6 and 3-7.

3. Transfer Function Data Entry

The input of transfer function data can best be explained by giving an example with a discussion of the topic. The coefficients for each polynomial of the transfer function (Equation 3-3) are contained within brackets and appear ordered from the smallest to the largest power of s. The distinction between numerator and denominator polynomials is made by the insertion of a division sign



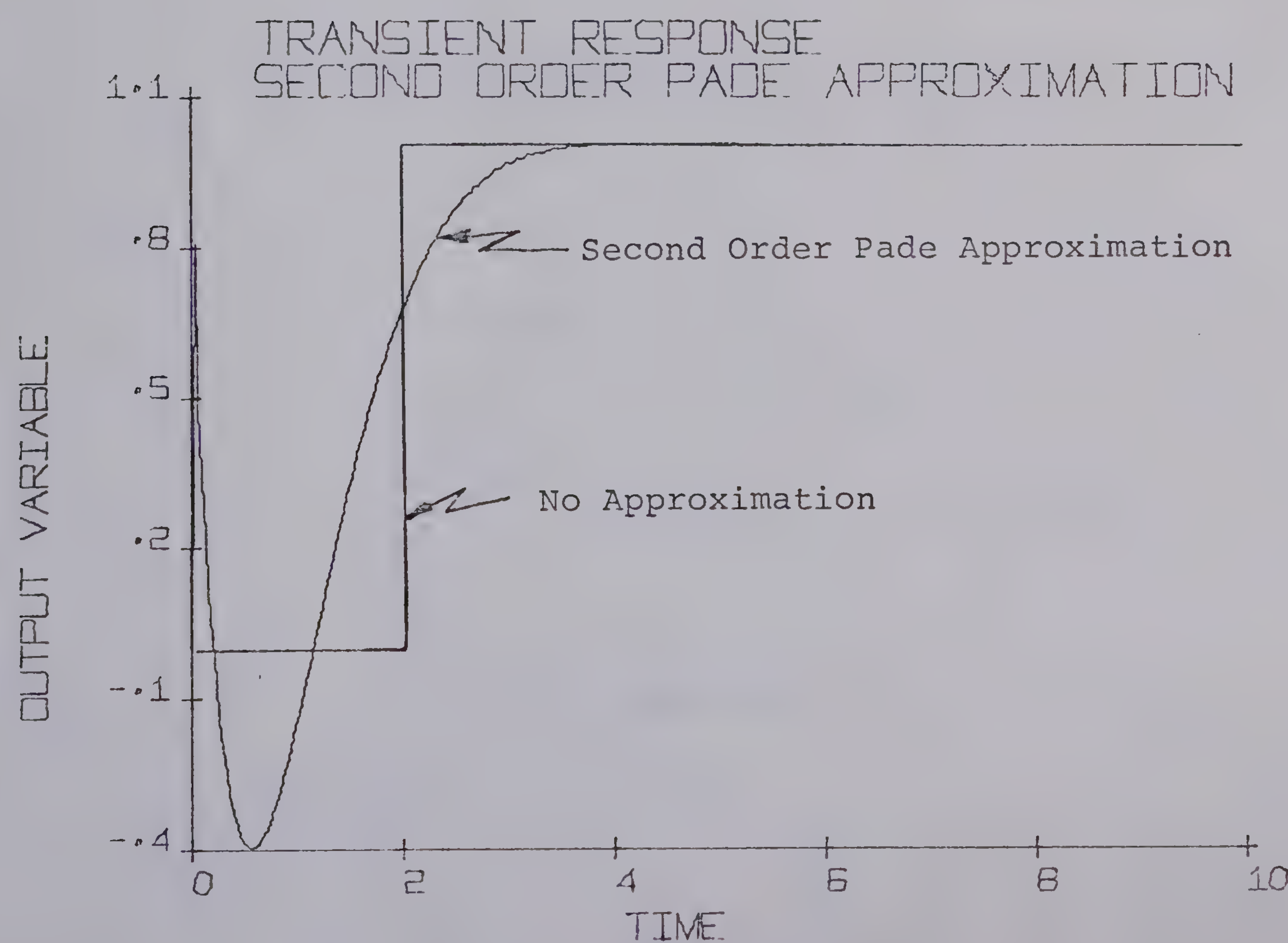
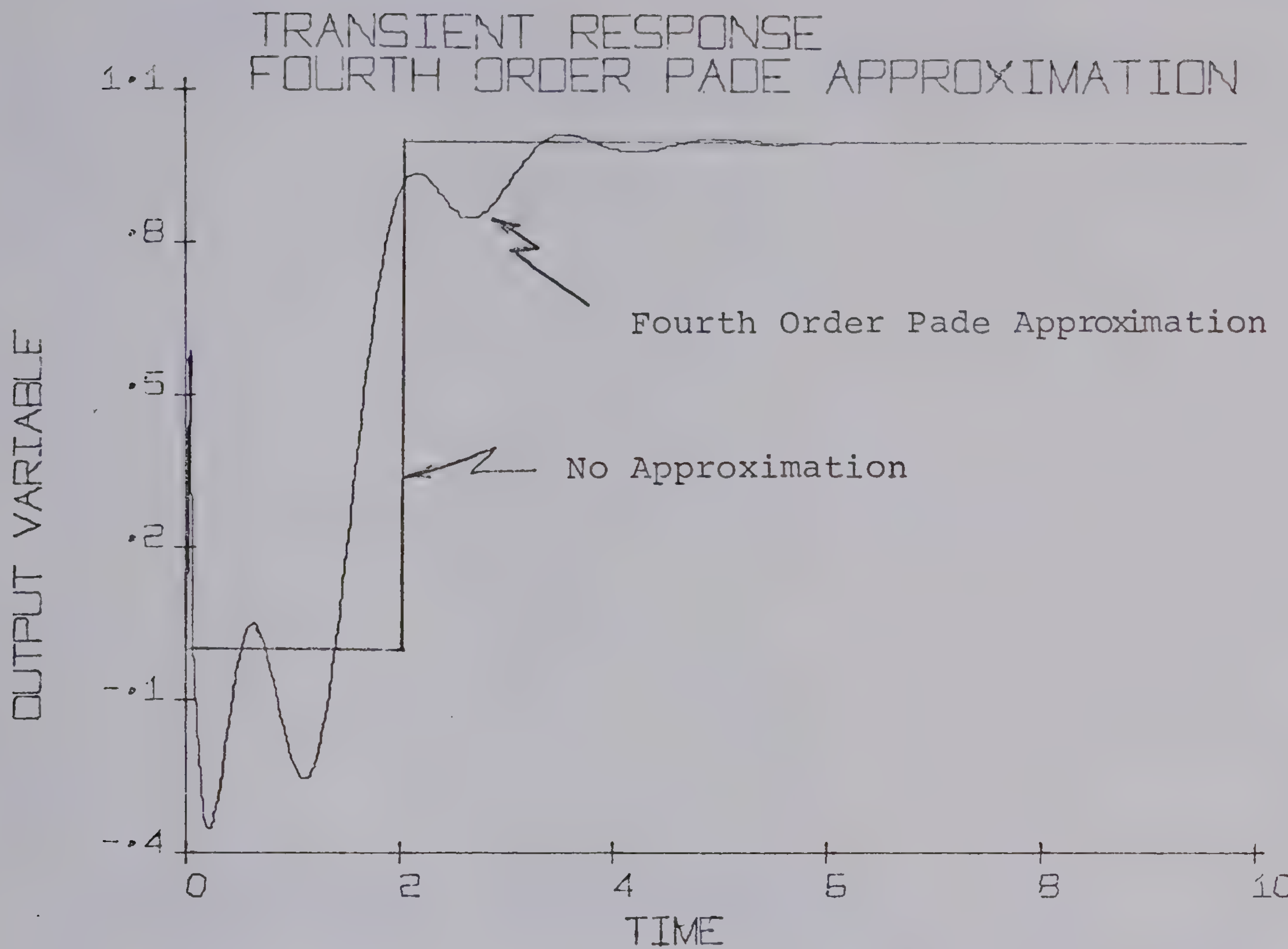


FIGURE 3-4 Pade Approximations for Time Delay



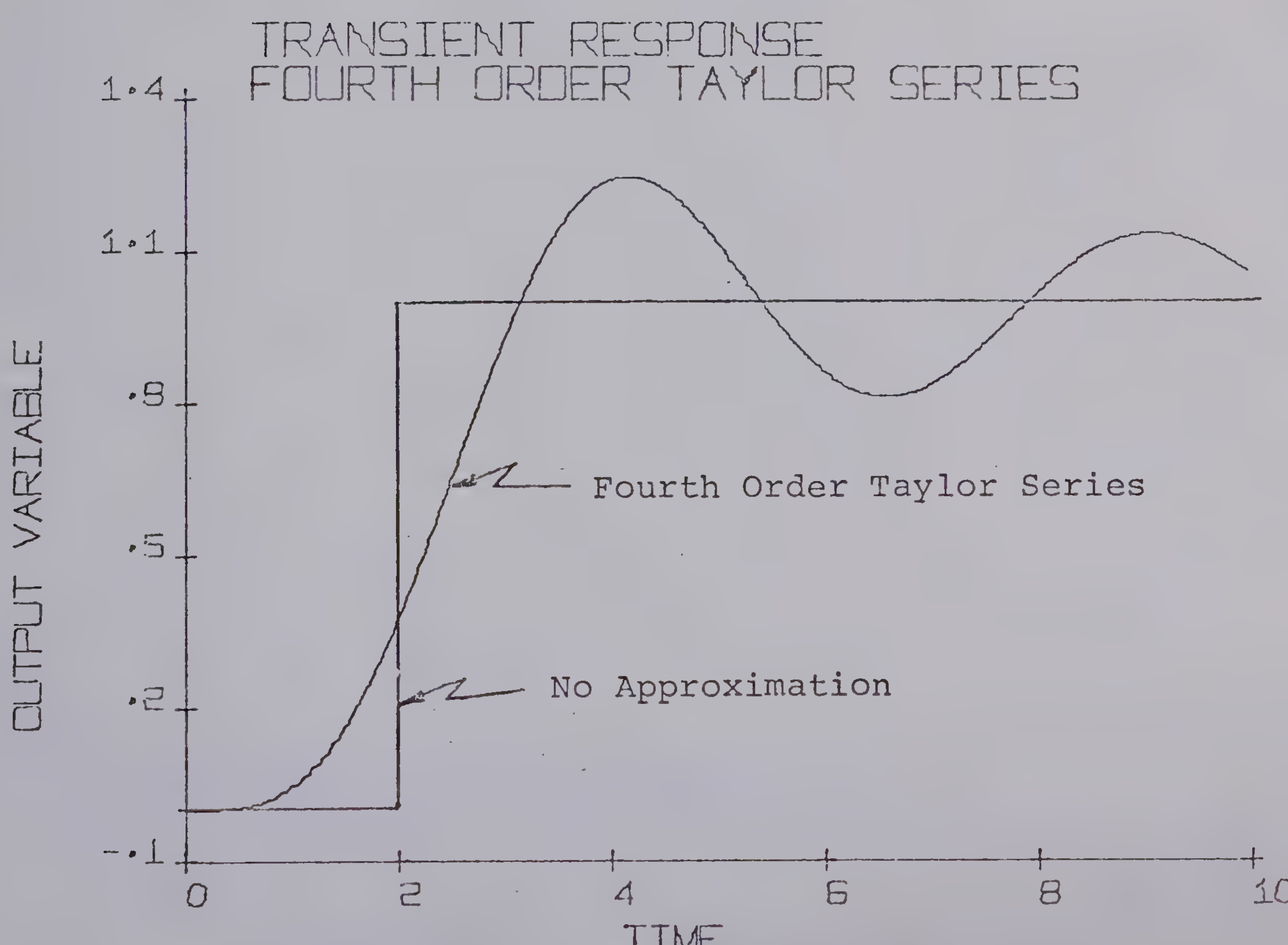


FIGURE 3-5 Fourth Order Approximations for Time Delay

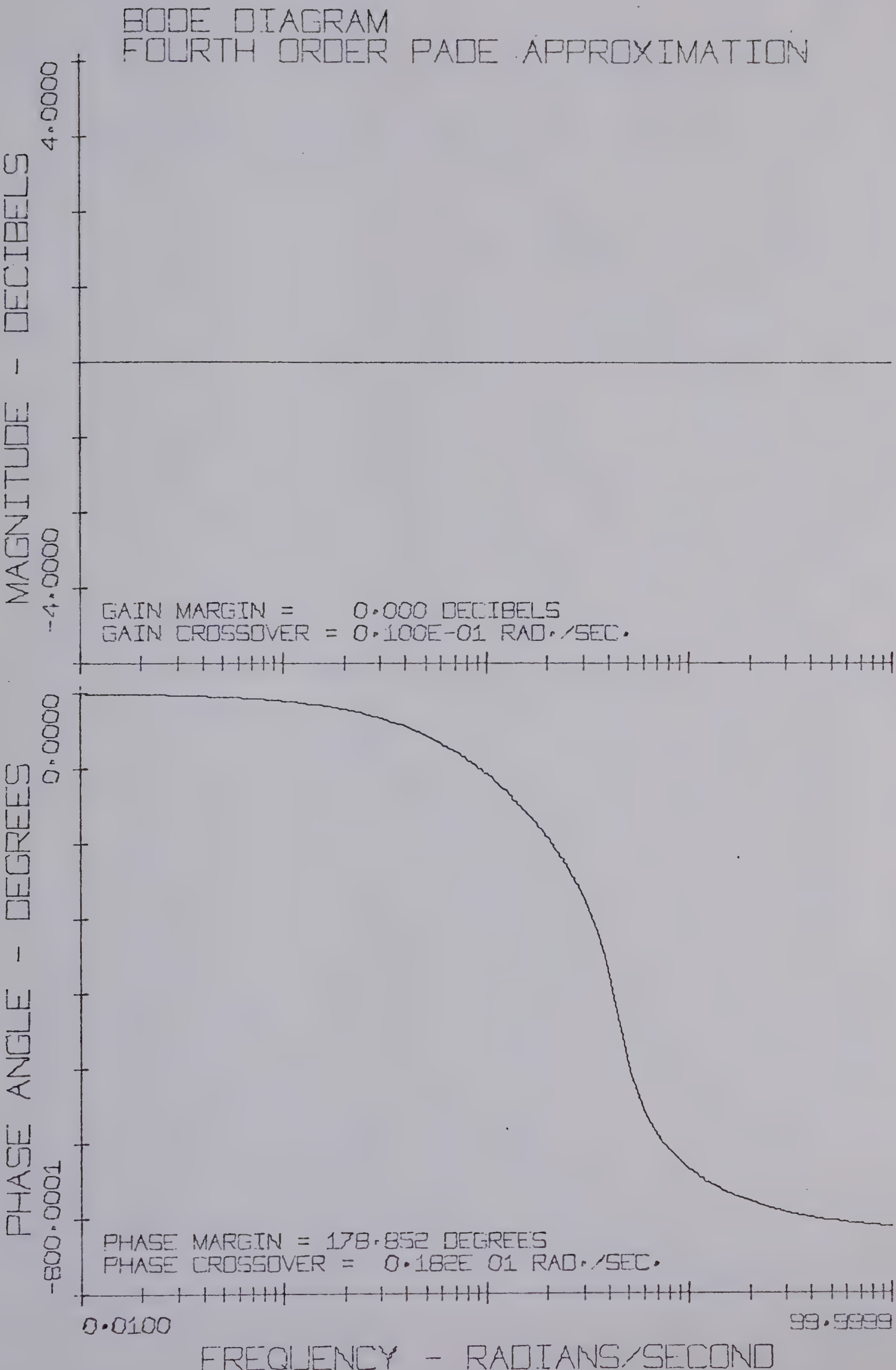


FIGURE 3-6 Bode Plot of Pade Approximation

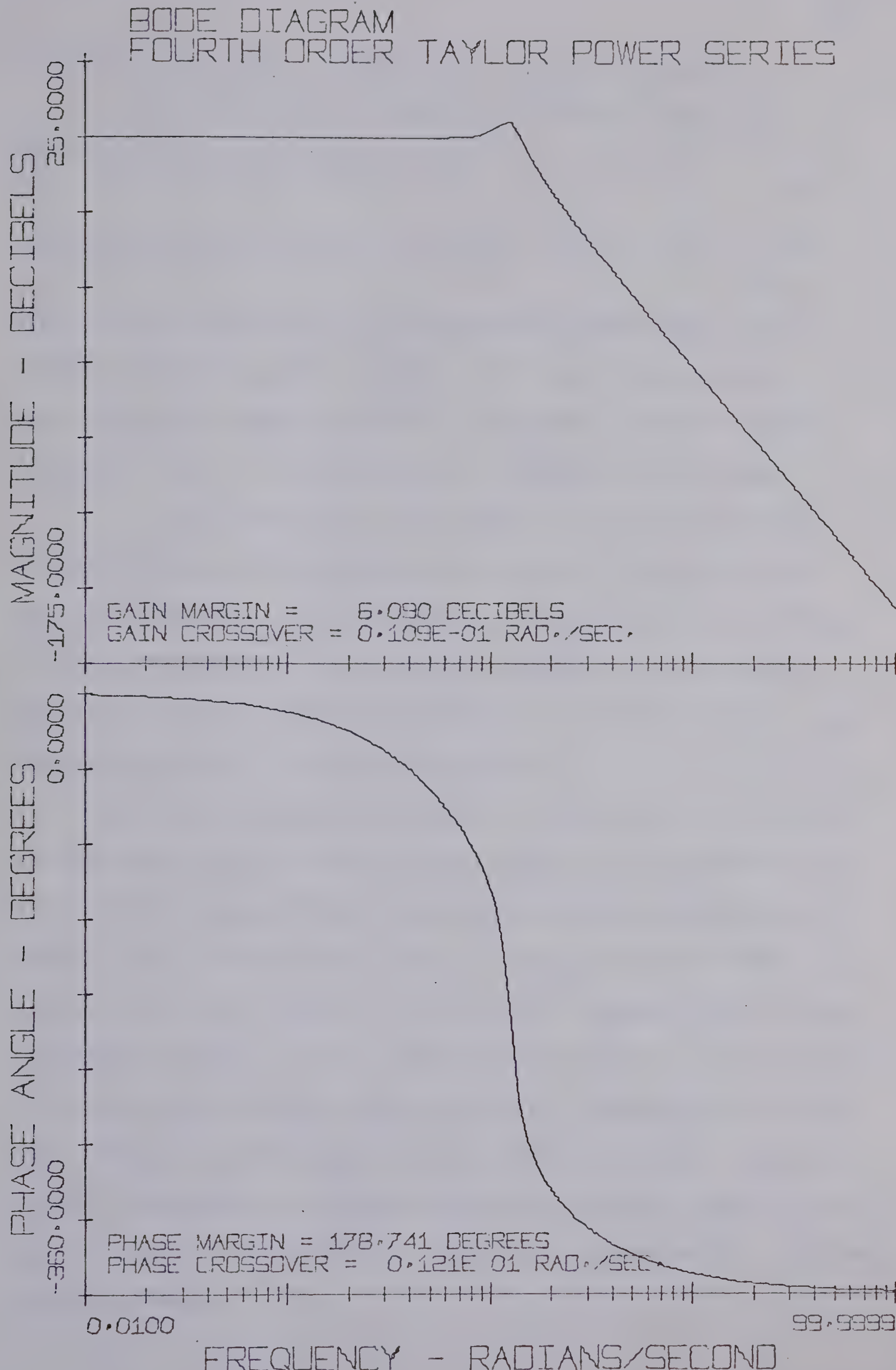


FIGURE 3-7 Bode Plot of Taylor Series Approximation

after all numerator polynomials have been entered.

The input for the transfer function given in Equation 3-3 would take the form:

$$P3\ (a_2,a_1)(a_5,a_4,a_3)(\ldots)/(b_2,b_1)(b_5,b_4,b_3)(\ldots) \quad (3\text{-}9)$$

This data is read in by a subroutine INPUT which performs a legality check on the first two alphanumeric characters to insure that they represent one of the nine blocks or 'TD' for time delay or transportation delay.

If the data passes this check, it is then passed to a subroutine GDFIB which will analyze the rest of it for the numerator and denominator polynomials. The data is then transferred by subroutine MDATA to the transfer function storage area and substituted for any earlier version of the P3 transfer function.

An error message is typed to the user if the entered transfer function name is not legal or if an error is found in the coefficient data upon further processing by GDFIB. For the transfer function name error the INPUT subroutine exits back to the executive program which then requests the next control digit to be entered. This therefore provides a convenient method for terminating the input section of the program. The pushing of the "end-of-field" character is interpreted as an illegal transfer functions name and is used for ending the inputting of transfer function data.

In contrast to the illegal name error an error in the input data is handled differently. Because the user has successfully entered a legal transfer function name the program assumes that he really wants to enter this transfer function. Therefore the program after typing an error message, requests that the data be entered again.

To configurate a typical control system the transfer functions of a number of the blocks must be entered into the system. The order in which these transfer functions is entered is irrelevant. An important point to keep in mind is that whenever a transfer function is entered correctly it replaces the earlier version of that transfer function. One exception to this was noted in this Chapter (Section C-2) where it was explained that the time delay approximation was multiplied by the transfer function previously entered for the P3 or H1 blocks of Figure 3-2.

4. <u>Storage of Transfer Function Data</u>

The transfer function data takes a form in storage similar to the input form shown in Equation 3-9. The polynomial data is stored as an array of coefficients ordered from the lowest to the highest power of s. Since the size of each numerator and denominator polynomial is not constant some form of data storage must be used so that the area used by each transfer function can float depending upon its own size. This means that a storage area equal to the largest transfer function will not be required for each transfer function of the system.

A method has been utilized that keeps the coefficients of all transfer function polynomials in a contiguous string in storage. It makes use of three of the blocks of COMMON IC(2, 40), ICONT (200), and DATA (400), as mentioned in Section B-3 of this Chapter. The IC array is used to store pointers for the other two arrays. The ICONT array stores the number of polynomials followed by the degree of each polynomial term that appears in the numerator or denominator of each of the transfer functions involved in the system. Similarly the DATA array contains the coefficients associated with each of these polynomial terms.

The system handled a total of twenty transfer functions composed of the nine block transfer functions of Figure 3-2 which the user enters and the rest such as the open and closed loop, the forward path, and the overall transfer functions being calculated by the program. There will therefore be forty numerator and denominator storage areas required.

The program associates a number ITF between one and twenty with each of the twenty transfer functions instead of using an alphanumeric representation similar to the user. The values of IC (1, 2*ITF-1) and IC (1,2*ITF) give the first entries in the ICONT array associated with the numerator and denominator respectively, of the ITF transfer function. The values of IC (2,2*ITF-1) and IC (2,2*ITF) point to the starting locations in the DATA

vector for the coefficients associated with the numerator and denominator polynomial terms respectively.

The method discussed above can be illustrated in the following example.

Block Transfer Function:

$$P2(s) = \frac{6.25\ (s + 10)}{s\ (s + 2)\ (s + 5)}$$

Data Entry to Program:

P2 (6.25) (10, 1)/(0, 1)(2, 1)(5, 1)

Storage in Arrays:

Assume that 1) P2 is the fourth transfer function

2) IC (1,7) = 15

3) IC (2,7) = 40

Then IC (1,8) = 18

IC (2,8) = 43

ICONT array starting at ICONT(15) contains

2, 0, 1, 3, 1, 1, 1

DATA array starting at DATA(40) contains

6.25, 10.0, 1.0, 0.0, 1.0, 2.0, 1.0, 5.0, 1.0

5. <u>Listing of Transfer Function Data</u>

Each transfer function can be listed automatically in the abbreviated form it will take in the storage array. This listing will be given if it has been entered correctly and, if the correct data switches have been set previously.

Another listing of this data can be obtained at any time by the entry of the CONTROL DIGIT 7. This option is used for checking previously entered transfer functions or for getting a complete listing of all the transfer functions for documentation purposes. The listing can be made on either the user's typewriter or the line printer by the subroutines LISTD and LSSTF. When a complete listing is made onto the printer, the program requests a title to be entered that will be associated with the listing.

The various forms that the transfer function listing can take will be illustrated in the Appendices.

6. Manipulation of Transfer Function Data

6.1 Initialization of the Transfer Function Data Arrays

The first time the program is executed or upon the user's command (CONTROL DIGIT = 8) an initialization function is performed on the transfer function area. A subroutine INTFA is executed which sets the order of numerator and denominator polynomial to zero. Each coefficient is made equal to a constant of 1.0. This means that all transfer functions would have a value of 1.0/1.0.

6.2 Inner Control Loop

There is one exception to this procedure. It is the case for the inner control loop whose numerator is initialized to a constant value of 0.0. This value, which means that there is no feedback for the inner control loop, remains until a different transfer function is entered for H2(s).

6.3 Open and Closed Loop Transfer Functions

The open loop and closed loop transfer functions of the overall system can be calculated once all the polynomial terms in the numerators and denominators of the block diagram transfer functions have been entered into storage. The routine OLOOP is executed first and performs a block diagram manipulation in a manner similar to that normally employed in block diagram simplification. The inner control loop is first checked for feedback. If it has feedback, it is converted to a single block in the forward path by the subroutine CANON. The transfer function calculated for the inner control loop is then multiplied by the other controller and process transfer function blocks to obtain the forward path transfer function which is then transferred to the storage area. The forward path transfer function is multiplied by the feedback path transfer function H1(s) to complete the calculation of the open loop transfer function which is also saved in storage.

The closed loop transfer function is calculated by the subroutine CLOOP. Previous to entering CLOOP another routine CLTYP is executed which requests the user to enter the type of response that he desires because the closed loop transfer function will be used to calculate the transient response. The program allows the user to substitute the open loop for the closed loop transfer function or to calculate the closed loop transfer function as the ratio of C(s)/R(s) or B(s)/R(s) as defined in Figure 3-2. This

option was included because although it is common to refer to C(s) as the output variable, this variable usually cannot be measured and the control signal is that from the measuring element transfer function H1, namely B(s).

If the closed loop transfer function B(s)/R(s) is desired, it is calculated by putting the numerator equal to the numerator of the open loop transfer function and the denominator equal to the sum of the numerator and denominator of the open loop transfer function. To calculate the closed loop transfer function C(s)/R(s) a slightly more complicated method must be used to manipulate the forward path and open loop transfer functions.

Extensive use of the polynomial manipulation routines of the Scientific Subroutine Package (17) was made in calculating the open and closed loop transfer functions. Two other subroutines were written for transferring transfer function data. GETTF determines a single transfer function for a particular block while TFMVE transfer a transfer function back to permanent storage.

CHAPTER IV

TRANSIENT RESPONSE CALCULATIONS

A. Laplace Transforms

Application of Laplace transforms to linear differential equations with constant coefficients yields an algebraic equation in another variable s. The solution of the differential equation is effected by simple algebraic manipulations in the s-domain. To obtain the desired time solution, it is necessary to invert the transform of the solution from the s domain back to the time domain. The Laplace transform method is used extensively and the pertinent theory can be found in many books on feedback control theory (11, 13, 22, 26, 29).

Two important advantages of the Laplace transform method are:

1) initial conditions are automatically incorporated into the solution for any arbitrary forcing function, and
2) solutions to problems involving simultaneous linear differential equations with constant coefficients can easily be obtained.

The usual practice in the analysis of control systems is to take the Laplace transformation of the original equations provided the initial conditions for all variables are zero. If the variables in the original equations do not

have zero initial conditions a change of variables is made before taking the Laplace transformation. The transformed equations are then combined and re-arranged and finally the inverse transformation of the final expression yields the desired solution for the transient response.

B. Inverse Laplace Transforms

The process of inverse transformation constitutes the major portion of the program for calculating the transient response. Even so, the procedure is orderly without using trial and error solutions, and the result yields both the steady-state, and transient solutions for given initial conditions.

The inverse Laplace transform may be evaluated by means of the inversion integral and the calculus of residues, but a much more expedient method is to use the partial fraction method.

1. Partial Fraction Method of Inverting Laplace Transforms

For simple transfer functions the inverse transforms are obtained by inspection. The more complicated transformed expressions require that a table of Laplace transforms be consulted to obtain the inverse transform function. If the expression whose inverse transform is desired is not in the particular table available, the complicated expression must be simplified into a sum of simpler

expressions, and the partial fraction method applied to perform the inversion.

The denominator of the overall algebraic transfer function F(s) must first be factored to yield an expression of the form shown in Equation 4-1.

$$F(s) = \frac{N(s)}{D(s)} = \frac{N(s)}{(s + s_1)\ (s + s_2)\ \ldots\ (s + s_n)} \qquad (4\text{-}1)$$

For an nth order denominator polynomial D(s), there must be n factors or roots. The values are the poles of F(s) or the values of s in the complex plane for which F(s) becomes infinite. These poles may be real, complex, or imaginary.

If all the poles are different, the partial fraction expansion takes the form

$$F(s) = \frac{N(s)}{D(s)} = \frac{K_1}{s + s_1} + \frac{K_2}{s + s_2} + \ldots + \frac{K_n}{s + s_n} \qquad (4\text{-}2)$$

where K_i, i = 1,2, ... n are constants. Once the K_i's are evaluated, the inverse Laplace transform may be taken term by term with the result being

$$F(t) = K_1\ e^{-s_1 t} + K_2\ e^{-s_2 t} + \ldots + K_n\ e^{-s_n t} \qquad (4\text{-}3)$$

Subsequent sections outline how to perform the partial fraction expansion for different pole combinations using a digital computer.

2. Calculation of Partial Fraction Coefficients

The response transfer function F(s) can be expressed as the ratio of two polynomials P(s) and Q(s). In this program, these polynomials must satisfy the following conditions:

i) the degree of P(s) must be lower than the degree of Q(s),

ii) these two polynomials must not have a factor in common,

iii) the coefficients of the two polynomials must be real numbers,

and iv) the coefficient of the highest power of s in the denominator must be equal to unity.

The program assumes the user is aware of these conditions and so does not explicitly check for these conditions.

The first step in performing the inversion by the partial fraction method is to factor Q(s) into linear and quadratic factors with real coefficients:

$$F(s) = \frac{P(s)}{Q(s)} = \frac{P(s)}{(s - s_1)(s - s_2) \ldots (s - s_k)} \qquad (4\text{-}5)$$

These values s_1, s_2, ... s_k are called the poles of F(s) but are also sometimes referred to as the zeros of the denominator. Now having determined the K poles of F(s) it is possible to express F(s) as a series of fractions, where the number of fractions is equal to k. If there are no repeated terms, the function F(s) can be expanded as

$$F(s) = \frac{P(s)}{Q(s)} = \frac{A_1}{s - s_1} + \frac{A_2}{s - s_2} + \ldots + \frac{A_k}{s - s_k} \qquad (4\text{-}5)$$

It is then necessary to evaluate the A constants and evaluate similar constants for repeated factors and complex factors of Q(s).

The Heaviside Expansion Theorems as given in references (7, 13, 33) are of great utility in systematizing the procedure of using partial fractions to invert the Laplace transforms. They outline how the coefficients for Equation 4-5 can be found for any pole of F(s) where it is real or part of a conjugate pair appearing in single or multiple order. There are four classes of problems encountered depending on the zeros of the denominator Q(s).

1) F(s) has first-order real poles
2) F(s) has repeated first-order poles
3) F(s) has a pair of complex conjugate poles (a quadratic factor in the denominator)
4) F(s) has repeated pairs of complex conjugate poles (a repeated quadratic factor in the denominator)

Each of the above cases will be discussed in some detail. In general, the Laplace transform F(s) will contain combinations of these cases. Many textbooks (7, 11, 13, 31) will give a more expanded description of the partial fraction method than will appear here.

For the first case Q(s) has a linear factor s - a that is not repeated. If $\phi(s)$ denotes the function which may be a quotient of polynomials that is left after removing the factor s - a from the denominator of F(s); such that,

$$F(s) = \frac{P(s)}{Q(s)} = \frac{\phi(s)}{s - a} \tag{4-6}$$

then according to the theory of partial fractions there exists a constant C such that

$$\frac{\phi(s)}{s - a} = \frac{C}{s - a} + H(s) \tag{4-7}$$

where H(s) represents the sum of the partial fractions that correspond to other linear and quadratic factors of Q(s) that appear once or are repeated. To evaluate the constant C, which is called the residue of F(s) at the pole $s = a$, multiply both sides of the above equation by the term $(s - a)$ and as $s \to a$ it is seen that $C = \phi(a)$. The inverse transform of the fraction $C/(s - a)$ is given by $\phi(a)\, e^{at}$. The function $\phi(a)$ can be evaluated from Equation 4-6 but $\phi(a)$ can also be calculated from the fact that $\phi(a) = P(a)/Q'(a)$. This is an important feature because all the constant C's corresponding to other linear non-repeated roots can be conveniently calculated from two polynomial evaluations for the numerator P(s) and the first derivative of the denominator Q'(s).

Considering the second case where the polynomial Q(s) contains a linear factor $(s - a)$ which is repeated n times. Following the same procedure as above it is possible to write

$$F(s) = \frac{P(s)}{Q(s)} = \frac{\phi(s)}{(s - a)^{n+1}} \tag{4-8}$$

where $\phi(s)$ is the quotient of polynomials obtained by removing the factor $(s - a)^{n+1}$ from the denominator of F(s).

There are now more terms necessary in the partial fraction representation of F(s). The form now taken will be as follows:

$$\frac{\phi(s)}{(s - a)^{n+1}} = \frac{A_o}{s - a} + \frac{A_1}{(s - a)^2} + \ldots + \frac{A_n}{(s - a)^{n+1}} + H(s) \tag{4-9}$$

where the constants A_r are not explicit functions of s and H(s) has the same meaning as in Equation 4-7. Again multiplying both sides by the repeated term results in

$$\phi(s) = A_o(s - a)^n + A_1(s - a)^{n-1} + \ldots + A_n + (s - a)^{n+1} H(s) \tag{4-10}$$

where it is seen that $A_n = \phi(a)$ as $s \to a$. To find the remaining n coefficients both sides of the above equation are differentiated $n - r$ times with respect to s to isolate the constant A_r. When $s \to a$ in the resulting equation, the relationship for the A_r's is found to be

$$A_r = \frac{\phi^{(n - r)}(a)}{(n - r)!} \tag{4-11}$$

The inverse transform of the r^{th} partial fraction is seen to be

$$\mathcal{L}^{-1}\left\{\frac{A_r}{(s - a)^{r+1}}\right\} = \frac{A_r\, r^r e^{at}}{r!} = \frac{\phi^{(n - r)}(a)\, t^r e^{at}}{(n - r)!\, r!}$$

(4-12)

Applying this relationship repeatedly as r goes from $0 \rightarrow n$ yields a summation of terms that will represent the inverse transform of $\phi(s)/s - a)^{n+1}$.

In the following section imaginary linear factors will be discussed. Since it was stated at the outset that the polynomials P(s) and Q(s) would have real coefficients, it follows that any imaginary linear factors of Q(s) resulted from a quadratic factor that could not be factored into two linear terms. The quadratic has the form $s^2 + \alpha s + \beta$, where $\alpha^2 - 4\beta < 0$ and therefore results in two complex conjugate factors. By completing the square, the original quadratic factor can be converted into a more useful form such as: $(s - a)^2 + b^2$ where a and b are real numbers and $s = a \pm ib$ are the complex factors of the original quadratic $s^2 + \alpha s + \beta$ also.

If the quadratic factor appears only once then the following relationship can be obtained

$$F(s) = \frac{P(s)}{Q(s)} = \frac{\phi(s)}{(s - a)^2 + b^2} = \frac{As + B}{(s - a)^2 + b^2} + H(s)$$

(4-13)

where A and B are real constants and H(s) has the same meaning as in Equation 4-7. Multiplying both sides of

this equation by the quadratic factor and letting $s \to a + ib$ gives

$$\phi(a + ib) = (a + ib) A + B \tag{4-14}$$

If ϕ_1 and ϕ_2 are said to represent the real and imaginary components of the complex number $\phi(a + ib)$ it can be seen that after suitable substitutions

$$F(s) = \frac{1}{b} \frac{(s - a)\ \phi_2 + b\,\phi_1}{(s - a)^2 + b^2} + H(s) \tag{4-15}$$

which has an inverse of the form

$$F(t) = \frac{1}{b}\ e^{at}\ (\phi_2 \text{ COS } bt + \phi_1 \text{ SIN } bt) + H(t) \tag{4-16}$$

Simplification of this expression can be achieved by converting to polar coordinates. It then would appear as

$$F(t) = \frac{1}{b}\ |\phi(a + ib)|\ \ e^{at} \text{ SIN } (bt + \Theta) \tag{4-17}$$

where Θ = argument of $\phi(a + ib)$ = ARCTANGENT $(\frac{\phi_2}{\phi_1})$

$$|\phi(a + ib)| = \sqrt{\phi_1^{\,2} + \phi_2^{\,2}}$$

A similar procedure to that for repeated linear factors must be followed in handling repeated quadratic factors. As the majority of the problems in the analysis of control systems can be handled by a quadratic factor that is repeated once, only this case will be considered.

The original Laplace transform can be expanded as

$$F(s) = \frac{(s)}{[(s - a)^2 + b^2]^2} = \frac{As + B}{(s - a)^2 + b^2} + \frac{Cs + D}{[(s - a)^2 + b^2]^2}$$

$$+ H(s) \tag{4-18}$$

The constants A, B, C, and D can be solved for by multiplying both sides of this equation by the square of the quadratic factor, differentiating with respect to s, and letting $s \to a + ib$. The term F(t) corresponding to the repeated quadratic factor can be written in the form

$$F(t) = \frac{1}{2b^3} \; e^{at} \; \{|\phi(a + ib)|[\text{SIN } (bt + \Theta) - bt \text{ COS } (bt + \Theta)]$$

$$- b|\phi'(a + ib)| \text{ COS } (bt + \Theta')|\} + H(t)$$

$$\text{where } \Theta' = \arg [\phi'(a + ib)] \tag{4-19}$$

3. Computer Implementation

In the previous section the theory of the method of partial fractions for calculating the inverse Laplace transform of a function F(s) was presented. The application of this theory in developing a program to solve for the time response from a function polynomial presentation in the s domain will now be discussed.

The transform function F(s) is a ratio of two polynomials P(s) and Q(s) as described in the previous section. In formulating a program to calculate the time response it was necessary to factor Q(s) into a set of linear and/or

quadratic terms. These terms or roots of Q(s) were calculated using the Newton-Bairstow method that is programmed into subroutine SOLUN.

The program that was developed to perform the inverse Laplace transformation will handle up to twelve linear factors each appearing once, twice or three times and up to eight quadratic factors each appearing once or twice. This is an arbitrary limitation restricted only by the dimensions of the storage arrays. The program calculates the coefficients for the four general cases as given in Equations 4-7, 4-9, 4-13 and 4-18. This information is then used by another subroutine described in the next section to calculate the time response.

There are two vectors of data associated with the inverse transform, one for the linear roots and the other for the quadratic terms. For the linear terms it is necessary to calculate A_o, A_o and A_1, or A_o, A_1 and A_2 of Equation (4-9) depending on the number of times the linear root is repeated. The calculation of the time response contribution made by linear roots can be simplified by using a modified version of Equation 4-11. Instead the following expression for the constants, shown in Equation 4-20, is used

$$A_r = \frac{\phi^{(n - r)}(a)}{(n - r)!\ r!} \qquad (4\text{-}20)$$

where n = number of times linear root (s - a) is repeated.

r = number of the coefficient to a maximum of n (Equation 4-9).

The time response function that results when the A_r constants are defined in this manner is shown below.

$$F(t) = e^{at} (A_o + A_1 t + A_2 t^2) + H(t) \qquad (4\text{-}21)$$

In determining the A_1 and A_2 constants using Equation 4-20, it is necessary to calculate the first and second derivatives of $\phi(s)$ and evaluate these derivatives at s = a. As stated before this function $\phi(s)$ could be a ratio of two polynomials in a form such as $\phi(s) = N(s)/D(s)$. By using the relationship for the derivative of a function u(s)/v(s), simplified forms for the derivatives can be formed as shown below.

$$\phi^{(1)}(s) = \frac{D(s)\ N^{(1)}(s) - N(s)\ D^{(1)}(s)}{D(s)\ D(s)}$$

$$\phi^{(2)}(s) = \frac{(D(s) - 2)}{D(s)}\ \frac{D(s)\ N^{(2)}(s) - N(s)\ D^{(2)}(s)}{D(s)\ D(s)}$$

The evaluation of the coefficients associated with a quadratic factor becomes slightly more difficult because the function must be evaluated for a complex number s = a + ib. For a non-repeated quadratic factor there

are four parameters necessary to completely define the time response as can be seen from Equation 4-22.

$$F(t) = C\, e^{at}\, \mathrm{SIN}\ (bt + \theta) \tag{4-22}$$

which is a simplified version of Equation 4-17 where $C = |\phi(a + ib)|/b$. For the case of a quadratic factor that is repeated once, Equation 4-23 shows that six parameters must be determined. This equation

$$F(t) = e^{at}\, \{C'(\mathrm{SIN}\ (bt + \theta) - bt\ \mathrm{COS}\ (bt + \theta)) + C''\ \mathrm{COS}\ (bt + \theta')\} \tag{4-23}$$

is the same as Equation 4-19 with some of the terms collected and given the following meaning

$$C' = |\phi(a + ib)|/2b^3$$
$$C'' = -|\phi'(a + ib)|/2b^2$$

4. Program Descriptions

The first step in performing the inversion from the "s" domain to the time domain is the calculation of the roots of the denominator of the overall transfer function of the system. A subroutine CALRT performs these calculations and begins by normalizing the coefficients of the denominator polynomial so the coefficient of the highest power of s is unity. It then calculates the linear and complex roots of the normalized polynomial by using

the Newton-Bairstow technique which was programmed into subroutine SOLUN.

After solving for the linear and quadratic roots, the coefficients of the inverse transform associated with each of these roots is calculated by subroutine INPFM. The coefficients for the linear roots are a, A_o, A_1 and A_2 and for the non-repeated quadratic roots a, b, c and θ while the repeated quadratic factor requires six parameters a, b, c, θ, C" and θ'. These coefficients were defined by Equations 4-20, 4-22, and 4-23.

Since many digital computers including the IBM 1800 do not have the complex arithmetic capability which is necessary for solving for the coefficients associated a complex root, three other subroutines were written for this function. They are PCMPY - to multiply two complex numbers together, PCDIV - to divide two complex numbers, and PCVAL - to evaluate a polynomial with real coefficients for a complex argument.

C. Forcing Functions

The main reason for calculating the closed loop transfer function is to allow the design engineer to calculate the effects that changes in one variable will have on the others in a unit or in a larger system made up of several component units. The changes of interest will generally be random variations - time functions of no special mathematical shape - such as outdoor temperature variations, or feed

stream purity or flow fluctuations.

Much valuable information about the effects of such changes can be obtained by considering the effects of a few standard changes. Some input functions which are frequently used for investigating the characteristics of a control system are the step, pulse, sinusoidal, ramp, or the impulse functions.

The procedure is to pick some property of the system which is subject to change directly by forces outside the system, and treat this as an independent variable or input. Some other property of the system will be the dependent variable of interest and may be considered the output. Once a transfer function F(s) is calculated that shows the relationship between these two variables, a disturbance can be introduced at the input to see its effect on the output. This is said to be forcing the system and the five standard input functions mentioned above are called the forcing functions.

Three variables, namely, the setpoint, load one, or load two, can be used as the input or study variable while the output variable can be either C(t) or B(t) as defined in Figure 3-2. The subroutine COFF requests the user to enter the variable to be studied, the type of forcing function to be applied, and any parameters associated with the forcing function such as step size, or pulse height and width. The choice of the output variable was previously obtained by subroutine CLTYP as discussed in Section 6.3

of Chapter 3. Once all this data is successfully entered, the subroutine OCLTF calculates an overall transfer function which may then be transferred to the programs used for calculating the inverse Laplace transformation and subsequently the time response.

D. Time Response of the System

1. Calculation of the Transient Response Data

The partial fraction coefficients must have been previously calculated before this Section of the program can proceed to calculate the time response data. An indicator is set any time the block diagram transfer functions are modified successfully. It is checked upon entering the time response calculation section of the system and causes an error message being outputted to the user if it was not reset. It is reset after the partial fraction coefficients are calculated.

The user is asked to supply data about the time response plot such as the end time, the number of data points to be calculated, and if it is an original plot or a replot of existing data. The time response will always be calculated from time $t = 0$ to the end time entered by the user if a new plot is requested.

The time response data is calculated from the partial fraction coefficients by subroutine STIME. It does this by a summation of the contributions of each linear and quadratic factor at every time interval. Up to a maximum

of five hundred time intervals are allowed by the program. The response data is stored into disk file immediately after calculation. The replot option allows the user to display this time response data later without recalculating it.

2. Display of the Transient Response Data

After the time response data has been calculated and stored onto disk, the subroutine DSPLT requests the user to enter additional information about the display desired. This input includes which display device is to be used, scope or plotter, the plot direction, and the title of the graph. There are four possible graph types that can be obtained. They are as listed below and represent the time response of the third order system

$$G(s)\ H(s) = K_c/(s + 1)(1/2\ s + 1)(1/3\ s + 1)$$

as K_c is varied and a unit step forcing function is applied to the setpoint.

i)	Vertical plot	(8½ x 11 inches),	Figure 4-1
ii)	Horizontal plot	(11 x 8½ inches),	Figure 4-2
iii)	Top Half plot	(8½ x 5½ inches),	Figure 4-3
iv)	Bottom Half plot	(8½ x 5½ inches),	Figure 4-3

3. Performance Criteria

The basic goal of a control system design is meeting performance specifications concerning the behavior of the system. Linear systems can be characterized by their

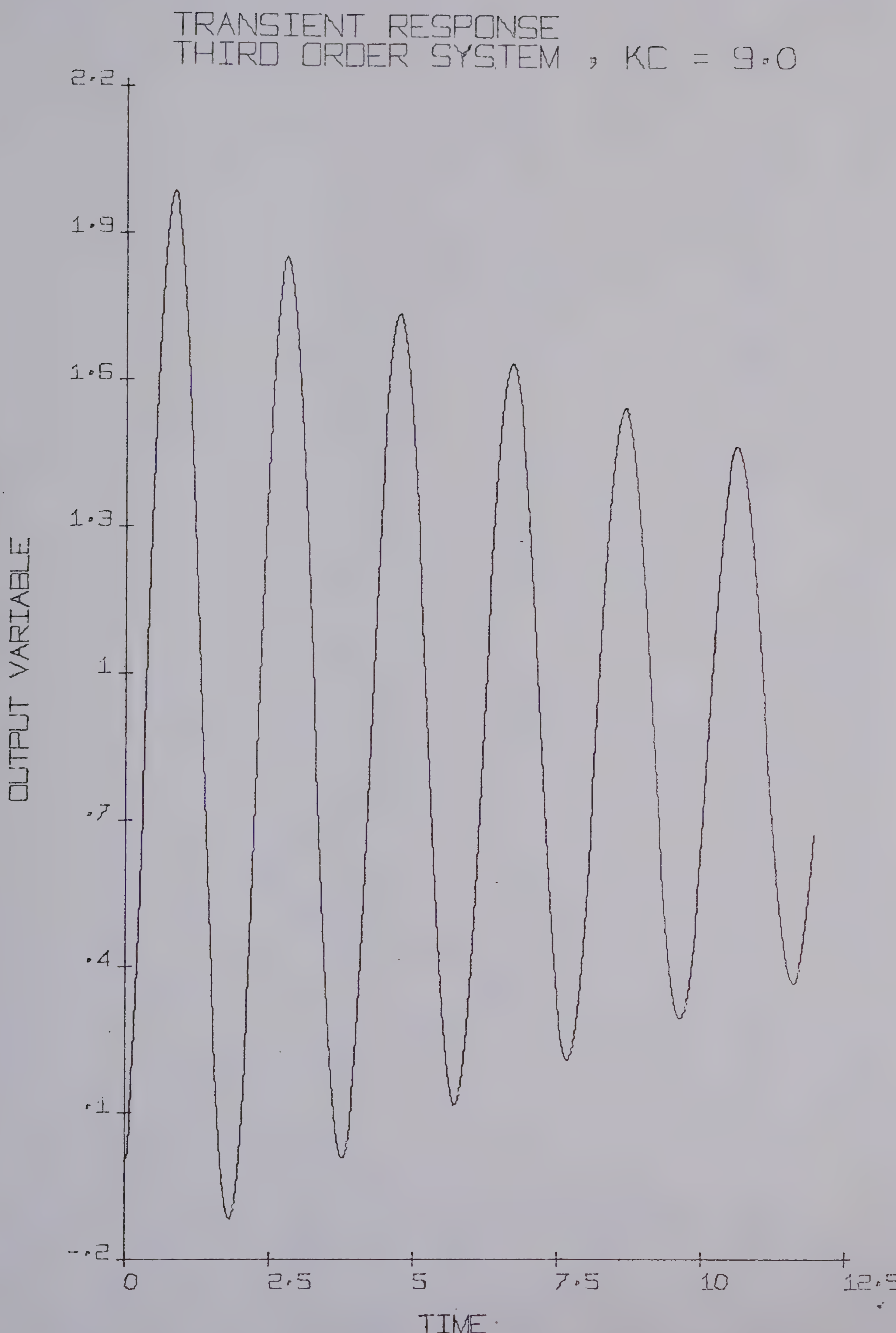


FIGURE 4-1 Example of a Vertical Plot

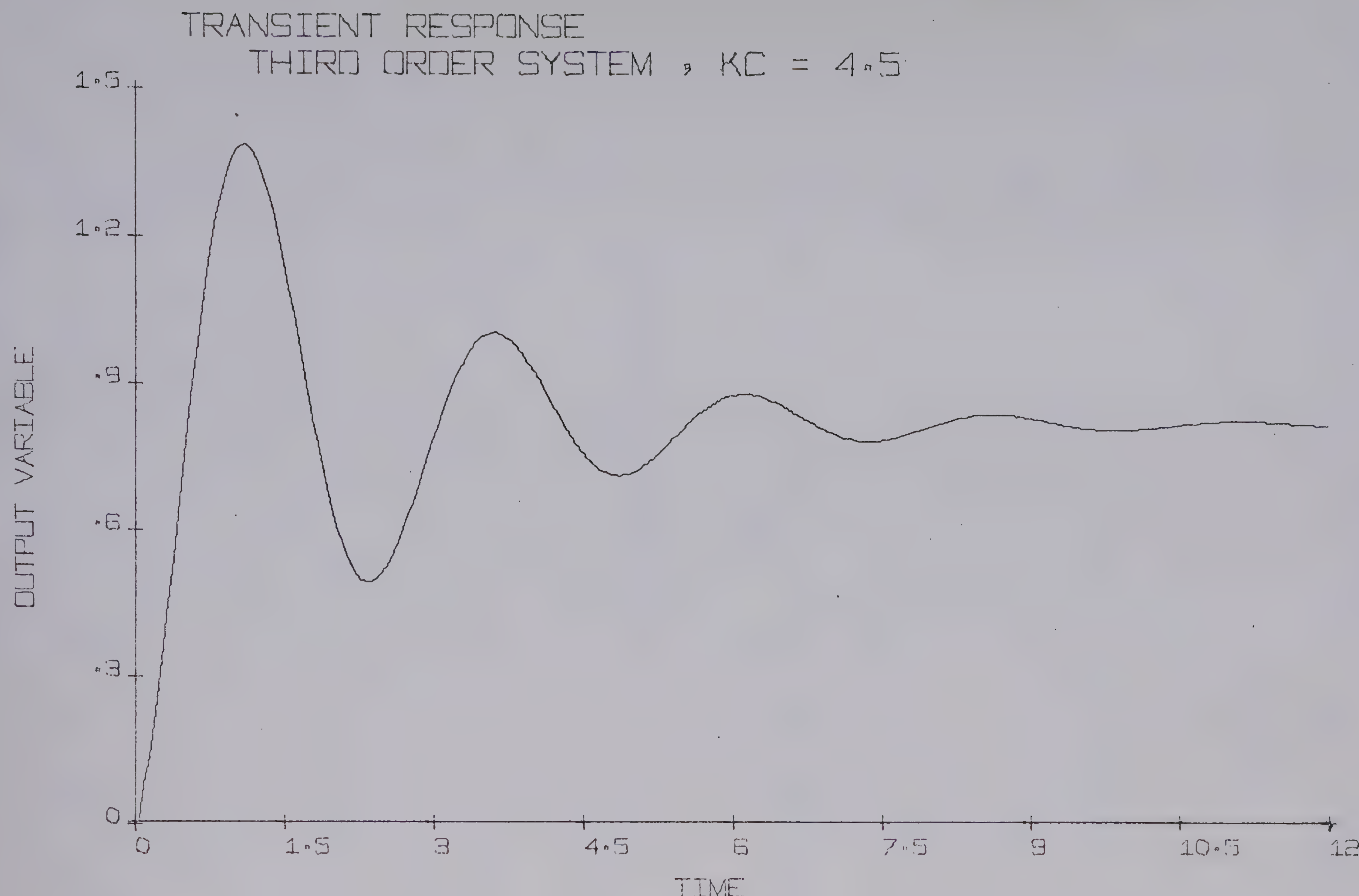
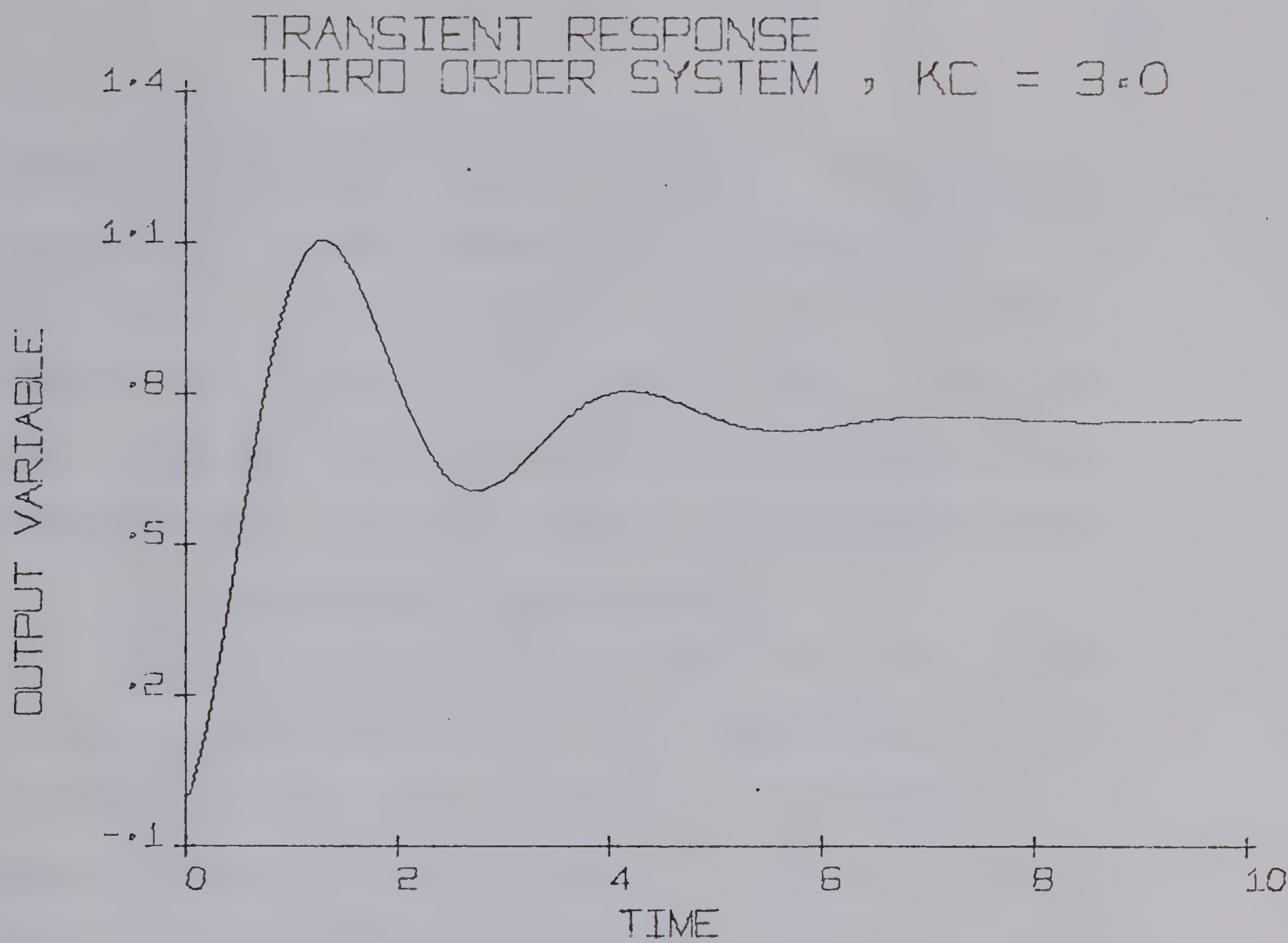


FIGURE 4-2 Example of a Horizontal Plot



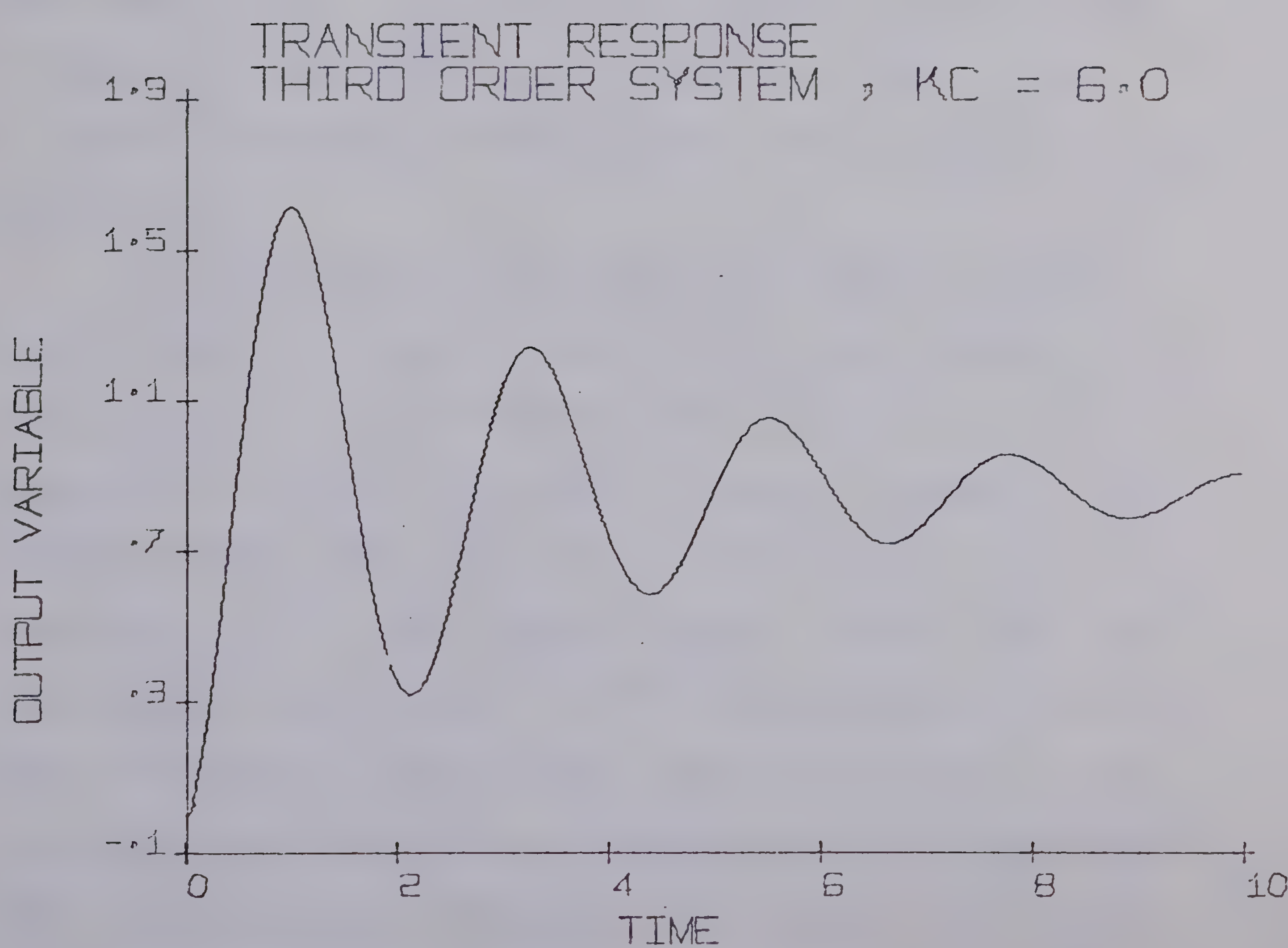


FIGURE 4-3   Example of Two Graphs Per Plot

response to aperiodic input functions. Consequently, the design of linear systems is often specified in terms of the transient response for simple aperiodic inputs. Very often a step function input is used and some of the commonly used time domain specifications associated with the step function are presented below.

3.1 SteadyState Calculations

The calculation of the steady state value SSTAT of the transient response has been implemented so two alternative methods are available to the subroutine SSCAL for determining its value. The first and simplest approach calculates the slope of the last few time response data values. If this slope is small, steady state is assumed to have been attained and an average value of the last five data values is used for the steady state value.

When the slope is not small, the second approach must be used. It results from the fact that complex conjugate roots of the characteristic equation give an exponentially damped sinusoidal transient response. An analysis showed that this was true for systems with a combination of real and complex roots. It was then stated that the curves SSTAT $\pm$ A e^{Bt} would comprise an envelope which would always contain the time solution. The subroutine SSCAL uses a least squares method to determine these unknowns using the peak values of the responses. This envelope has been drawn onto Figure 4-4.

3.2 Transient Response Criteria

The transient response is normally analyzed by using a unit-step input as the forcing function. The response can then be characterized by the following:

i) Percentage Overshoot - the percentage overshoot is found from the amount that the transient response to a step input exceeds the final or steady state value of the output. It is usually expressed as a percentage of the final value and is a measure of relative stability. Allowable values for the peak overshoot range up to 25 or 30 percent if the settling time is fast. Overshoots higher than this are seldom acceptable.

ii) Peak Time - the time required by the normal response or output to reach its first maximum.

iii) Rise Time - the rise time is defined (26) as the time required for the response to a unit step function input to change from 10 to 90 percent of its final value. It gives a measure of the system response speed.

Also included as one of the time domain specifications and similar to the rise time is the time required for the normal response to the step input to reach its final value for the first time.

iv) Delay Time - the usual definition of the time delay is that it is the time interval between the excitation and a measurable response. As a time

domain specification it is often defined as the time required for the response to a unit step to reach fifty percent of its final value and this is the time calculated by the program.

v) Decay Ratio - this is the ratio of successive overshoots in the time response.

vi) Settling Time - the time for the response to reach and remain within a specified percentage of the final value after a step forcing function is applied. Values of two to five per cent are often quoted as the specified percentages. The most commonly used value is five per cent so the program has been written using this value.

3.3 Error Integral Criteria

The purpose of a feedback control system is to minimize the output error as a function of time after some upset has been introduced into the system. Both the magnitude of the error and the time over which an error exists should contribute to the definition of optimum control. Three of the most common error-integral criterions of performance are discussed below and are available on the system.

i) IAE - Integral of the Absolute Value of the Error

$$IAE = \int_0^\infty |e(t)| \, dt$$

This criterion increases for either positive or negative errors. It is more sensitive to small errors than the ISE.

ii) ISE - Integral of the Squared Error

$$ISE = \int_{o}^{\infty} e^{2}(t)\, dt$$

Both positive and negative values of the error increase the size of the integral. The ISE criterion usually yield responses that have small overshoots but long line-out times since small errors contribute little to the integral.

iii) ITAE - Integral of Time Multiplied by Absolute Error.

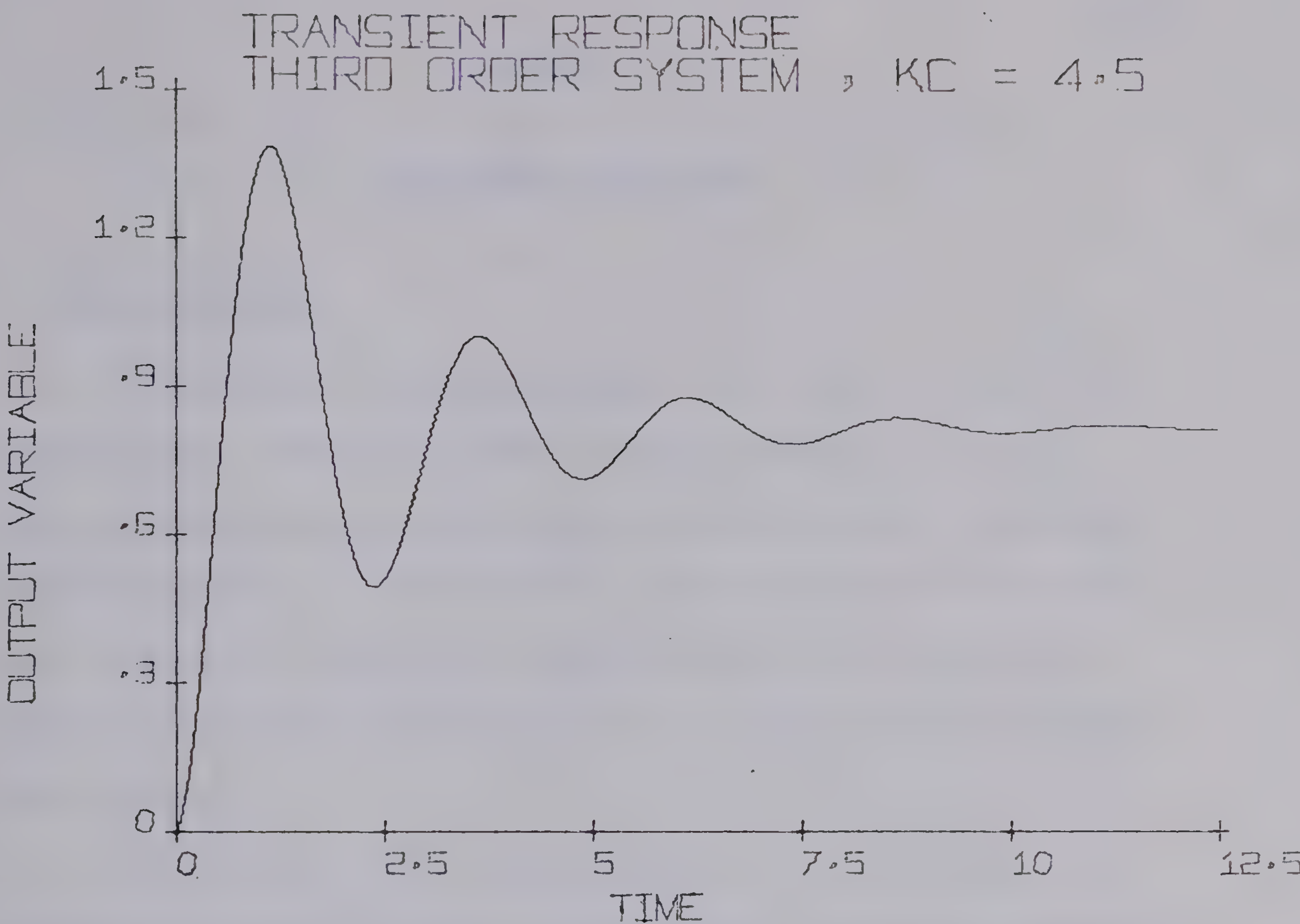
$$ITAE = \int_{o}^{\infty} t|e(t)|dt$$

This criterion is insensitive to the initial errors but weights heavily errors occurring late in time. This means that the response will show a shorter total response time with less oscillation than the other criteria but it may exhibit larger overshoots initially.

3.4 Display of the Performance Criteria

Because the performance criteria that were discussed above were so closely associated with the time response of the system, it was decided that it should be displayed to the user with the transient response. The programs that calculate these criteria allow the user to show the results on the scope or plotter as the bottom plot of a two figure plot as shown in Figure 4-4. Another option allows the

performance criteria to be typed out on the user's console. This might be desirable when hard copy is required but plots of the time response have already been obtained or are not required.



PERFORMANCE CRITERIA FOR ABOVE RESPONSE

STEADY STATE	=	0.8144
PERCENT OVERSHOOT	=	70.04
PEAK TIME	=	1.1272
RISE TIME 0 TO 100	=	0.5543
RISE TIME 10 TO 90	=	0.3907
DELAY TIME	=	0.3366
DECAY RATIO	=	0.3317
SETTLING TIME	=	6.5130

ERROR INTEGRALS

IAE	=	0.02036
ISE	=	0.01656
ITAE	=	0.01016

FIGURE 4-4 Display of Performance Criteria

CHAPTER V

FREQUENCY RESPONSE

A. Introduction

Although the time response of a system to a function of time impressed upon its input terminal is of basic importance much can be learned from the frequency domain analysis. The essential feature of the frequency-domain method is that the description of the system is given in terms of its response to a sinusoidally varying input signal.

The steady state response of a linear system to a sinusoidal input is also a sinusoidal of the same frequency. Therefore, the effect of the system upon the signal transmitted through it can be described in terms of two quantities:

a) The ratio of the amplitude of the response sinusoid to the amplitude of the input sinusoid, and

b) The phase of the response sinusoid with respect to the input sinusoid.

The amplitude ratio is termed the gain of the system, and the relative phase angle, positive if the response leads the input and negative if the response lags the input, is the phase angle of the system.

An important feature of the frequency response method is that the transfer function describing the sinusoidal

steady-state behavior of the system can be obtained from the transfer function simply by replacing the Laplace operator s by $j\omega$. The transfer function is thus a function of complex variables, and, in general, can be represented by the gain and the phase angle, each a function of the frequency. The complete frequency response is therefore revealed by presenting both gain and phase angle over the frequency range $0 \leq \omega \leq \infty$.

Frequency response methods are based on the open-loop transfer function G(s) H(s) as given in Figure 3-1. The plots are made by replacing s by $j\omega$ in the transfer function, G(s) H(s). The performance of the closed-loop system is predicted from these plots.

The following method of plotting the transfer function $G(j\omega)$ $H(j\omega)$ are the most useful in the frequency response analysis and design of feedback control systems:

a) Bode Plot (Corner Plot): A plot of the magnitude (in decibels) versus the log of the frequency on rectangular (or semilog) coordinates and another plot of the phase angle versus the log of frequency also on either rectangular or semilog coordinates.

b) Nyquist Plot (Polar Plot): A plot of the magnitude versus phase angle on polar coordinates as the frequency is varied from zero to infinity.

c) Log-Modulus Plot (Gain Phase Plot): A plot of the magnitude (in decibels) versus the phase

angle of $G(j\omega)\ H(j\omega)$ on rectangular coordinates with frequency as a varying parameter on the curve.

These graphs and the procedure by which they were obtained will be discussed in further detail in the following Sections.

B. The Bode Plot (Corner Plot)

The frequency response of the system can be represented by a plot of the magnitude (gain) versus the frequency and a plot of the phase angle versus the frequency. The Bode plot is also known sometimes as a corner plot or simply the logarithmic plot of the open loop transfer function.

The unit of gain most often used in Bode plots is the decibel which is defined as twenty times the common logarithm to the base ten of the magnitude. To facilitate the presentation of the frequency response characteristics over a large range of frequency, a logarithmic scale is used on the frequency axis. The phase angle is also plotted using semi-logarithmic coordinates.

Since the decibel is a logarithmic measure of the gain, the over-all gain in decibels of two components cascaded in series can be found by adding the gains in decibels of the individual components. The fact that the effect of cascading components can be determined by graphical addition in the Bode diagram, rather than by graphical

multiplication as in the use of the Nyquist plot is a significant advantage.

To illustrate this method of exhibiting frequency-response characteristics, a Bode plot is given in Figure 5-1 for the following transfer function:

$$G(s)\ H(s) = \frac{10\ (s + 10)}{s\ (s + 5)(s + 2)}$$

It should be noted that in addition to the Bode plot in this figure, the gain and phase margins and crossover frequencies have been included to present as much infomation about the system as possible. An optional listing of the frequency response data can be made on the line printer either when it is calculated or replotted. This will not be illustrated here but an example of the output may be found in the User's Manual (Appendix A).

C. The Nyquist Diagram (Polar Plot)

The polar plot of a transfer function $G(s)\ H(s)$ is a plot of the magnitude versus the phase angle $G(j\omega)$ $H(j\omega)$ on polar coordinates as ω is varied from zero to infinity. From a mathematical viewpoint, it is regarded as a mapping of the positive half of the imaginary axis of the plane of the complex variable s onto the plane of the function $G(j\omega)\ \ H(j\omega)$.

The Nyquist diagram is obtained by expressing the frequency function at a given frequency as a vector in

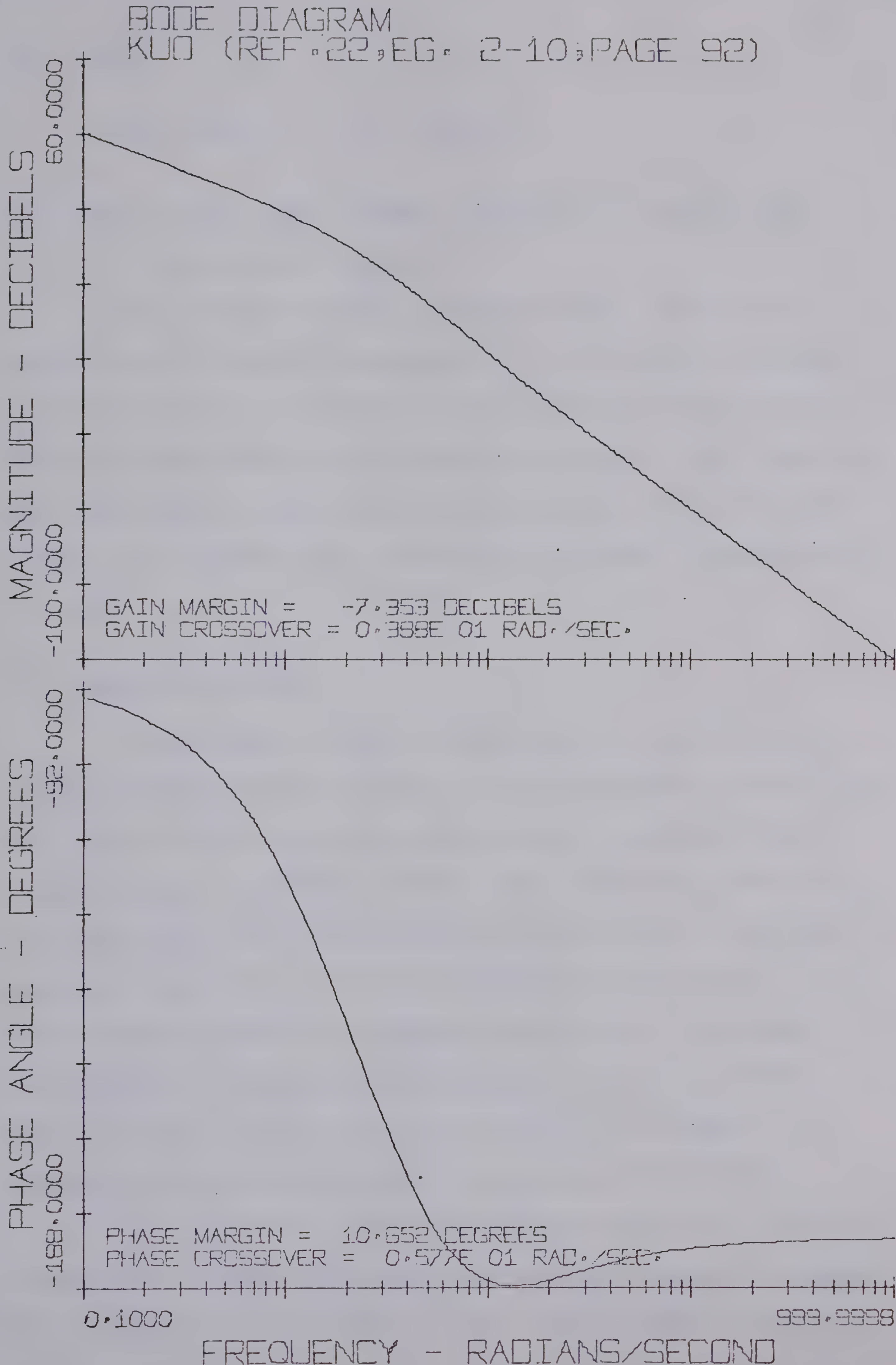


FIGURE 5-1 Example of a Bode Diagram

polar form:

$$G(j\omega)\ H(j\omega) \;=\; K(\omega)\ \underline{/\ \phi(\omega)}$$

The locus of the tips of these vectors as ω varies from 0 to ∞ is the Nyquist diagram.

The subroutine NYPLT allows the user a choice of scales for the Nyquist diagram. It can be either automatic where the computer chooses the starting and ending coordinates for both axes by the subroutine COSGR or the user can specify these, thereby limiting the graph to a particular region. An illustration of these two methods, respectively, is given in Figure 5-2 and 5-3.

D. Log-Modulus Plot

As mentioned in the introduction of this chapter, the log-modulus plot is a plot of the transfer function G(s) H(s) in decibels versus its phase angle in degrees with frequency as a parameter on the curve. The main advantage of using this set of coordinates is that such a plot can be superimposed on the Nichols chart to determine the relative stability and the frequency response of the closed-loop system. When the gain constant K of the transfer function $G(j\omega)\ H(j\omega)$ is varied, the plot is raised or lowered vertically according to the logarithmic scale.

The magnitude versus phase plots are usually obtained by first plotting the Bode plot and then using the same data to plot the log-modulus plot. The program follows a

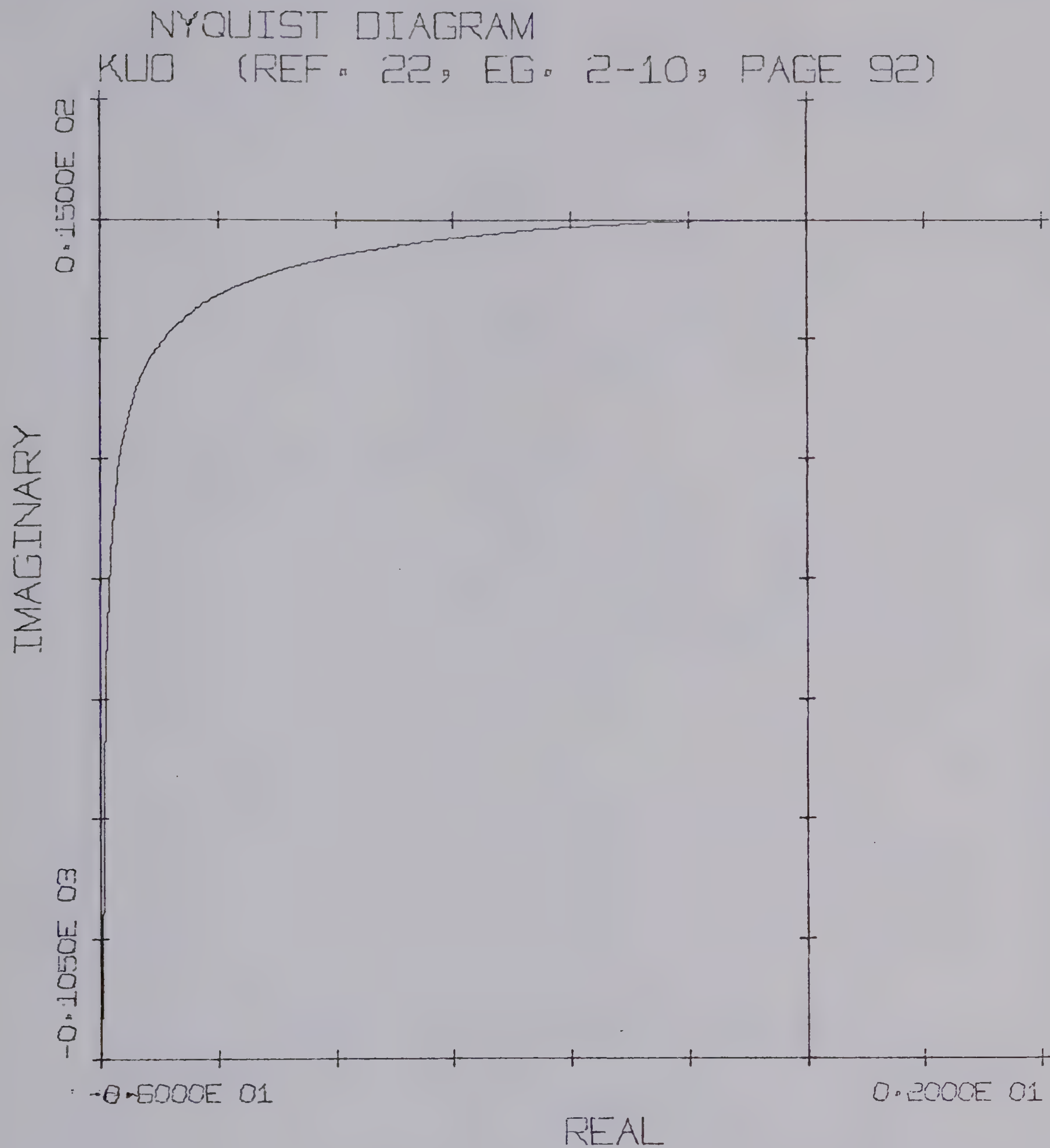


FIGURE 5-2   Nyquist Diagram Automatic Scaling

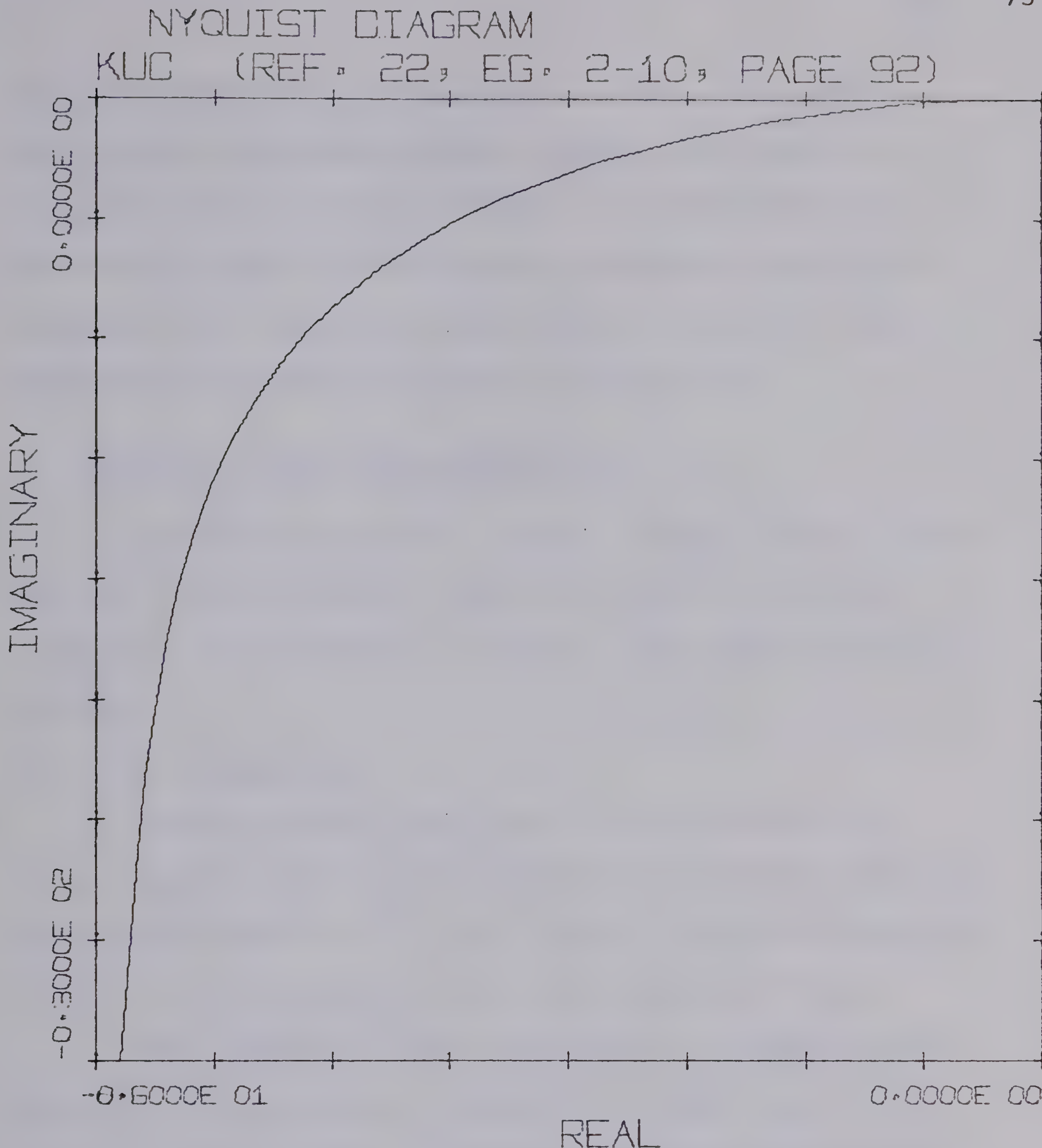


FIGURE 5-3   Nyquist Diagram - User Specified Scaling

similar approach by calculating the data for a Bode diagram and then plotting the data onto the log-modulus plot by subroutine LMPLT. In addition to the log-modulus graph, some frequency response specifications are also listed at the bottom of the plot. An example of this plot is shown in Figure 5-4.

E. Frequency Domain Specifications

The performance of a control system can be characterized in the frequency domain by several different criteria. A description of some of the common criteria follows:

1. Bandwidth

The bandwidth is defined as the frequency at which the magnitude of the closed loop transfer function $M(j\omega)$ has dropped to 70.7 per cent of its zero frequency level, or 3 db. down from the zero frequency level.

The bandwidth gives the range of frequencies for which the system gain is adequate to insure good transmission of the signal. A large bandwidth usually indicates that higher frequency signals will be passed on to the output. Thus, the transient response may have a faster rise time accompanied by a larger overshoot. Conversely, if the band width is small, only low-frequency signals are passed; therefore, the time response will generally be slow and sluggish.

2. Peak Resonance, M_p

This is defined as the maximum value of the closed loop transfer function $M(j\omega)$ and also gives an indication of the relative stability of the system. The peak resonance has been shown to be related to peak overshoot for second order transfer functions (13, 22, 26) and any system determined by a single pair of conjugate poles could be satisfactorily designed using the M_p as a specification. It is apparent from Figure 5-27 given by Murphy (26) that a high M_p corresponds to a large overshoot in the time response. For design purposes, it is generally accepted that an optimum value of M_p is 1.1 to 1.5 which results in an overshoot of from 15 to 30 per cent.

3. Resonant Frequency, ω_p

This is the frequency at which the peak resonance M_p occurs.

4. Gain Margin

The gain margin is a measure of the closeness of the phase-crossover point to the critical point. The gain margin frequency is first calculated as the frequency at which the phase crossover occurs (phase angle equals -180 degrees) and then the gain margin is found by determining the magnitude in decibels at the gain margin frequency.

The physical significance of gain margin is usually that it is the amount of gain in decibels that can be allowed to increase in the loop before the system reaches

instability. The gain margin must be positive when expressed in decibels for a stable system.

5. Phase Margin

The phase margin is a measure of the closeness of the gain-crossover point to the critical point. It is calculated by finding the phase angle of $G(j\omega)H(j\omega)$ at the gain crossover frequency (point where the magnitude is 0 decibels) and then adding 180 degrees. The phase margin is the amount of phase shift at the gain crossover frequency that would just produce instability. The phase margin must be positive for a stable system and a commonly used value for design purposes is 45 degrees.

It is usually preferable to evaluate the gain and phase margins of a control system from its Bode plot. The reason is simply that the Bode plot is easy to construct, and the margins can be obtained by inspection. Because this was true for the cases when the Bode plot was drawn by hand, the frequency analysis programs calculate the gain margin and phase margin and display the values on the Bode plot. As an illustrative example, consider the open-loop transfer function of a unity feedback system.

$$G(s) = \frac{K}{s\,(1 + 0.2s)(1 + 0.02s)}$$

The Bode plot is shown in Figure 5-5 with the log-modulus plot (Figure 5-6) giving the additional specifications of peak resonance and resonant frequency.

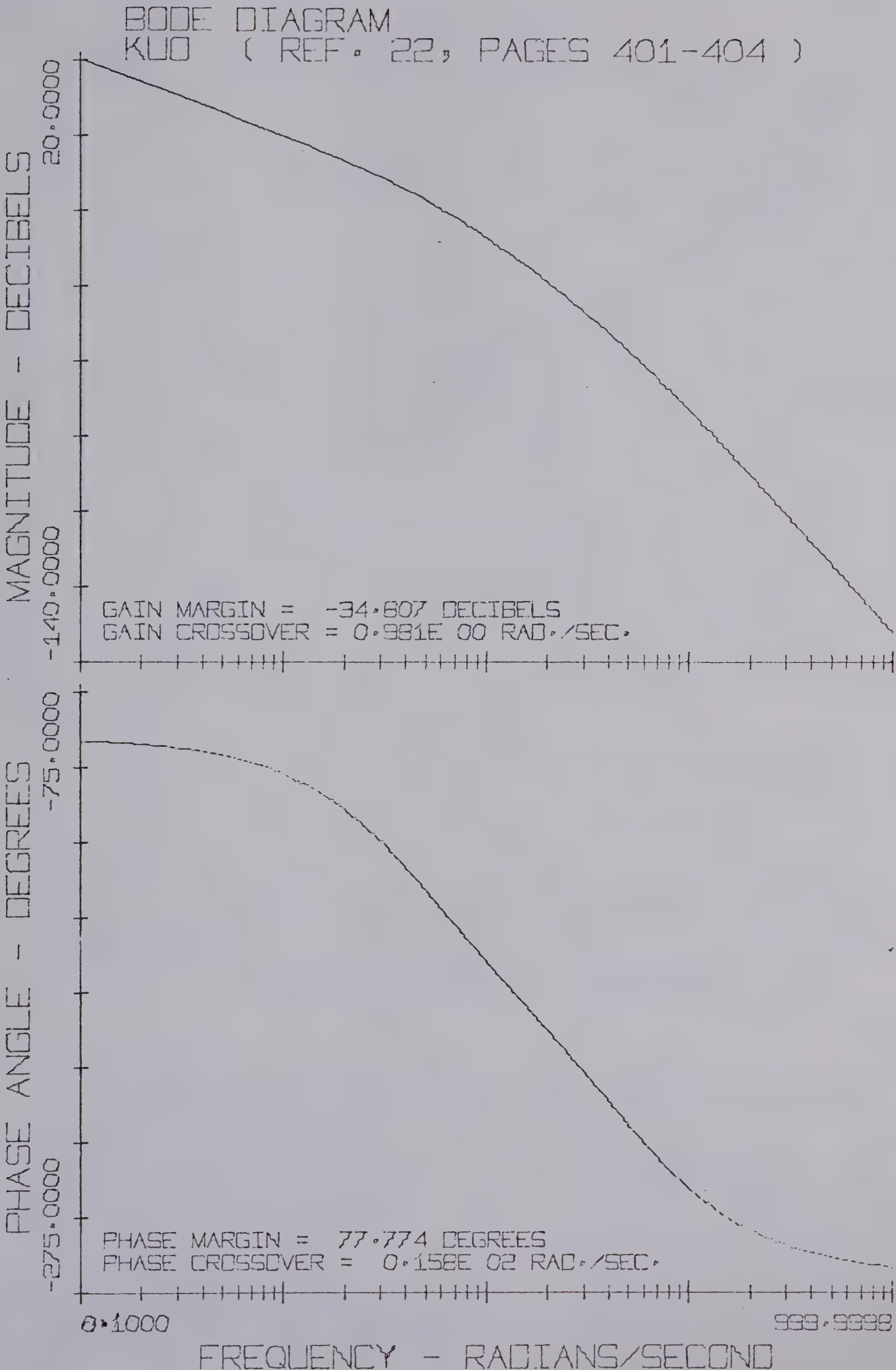
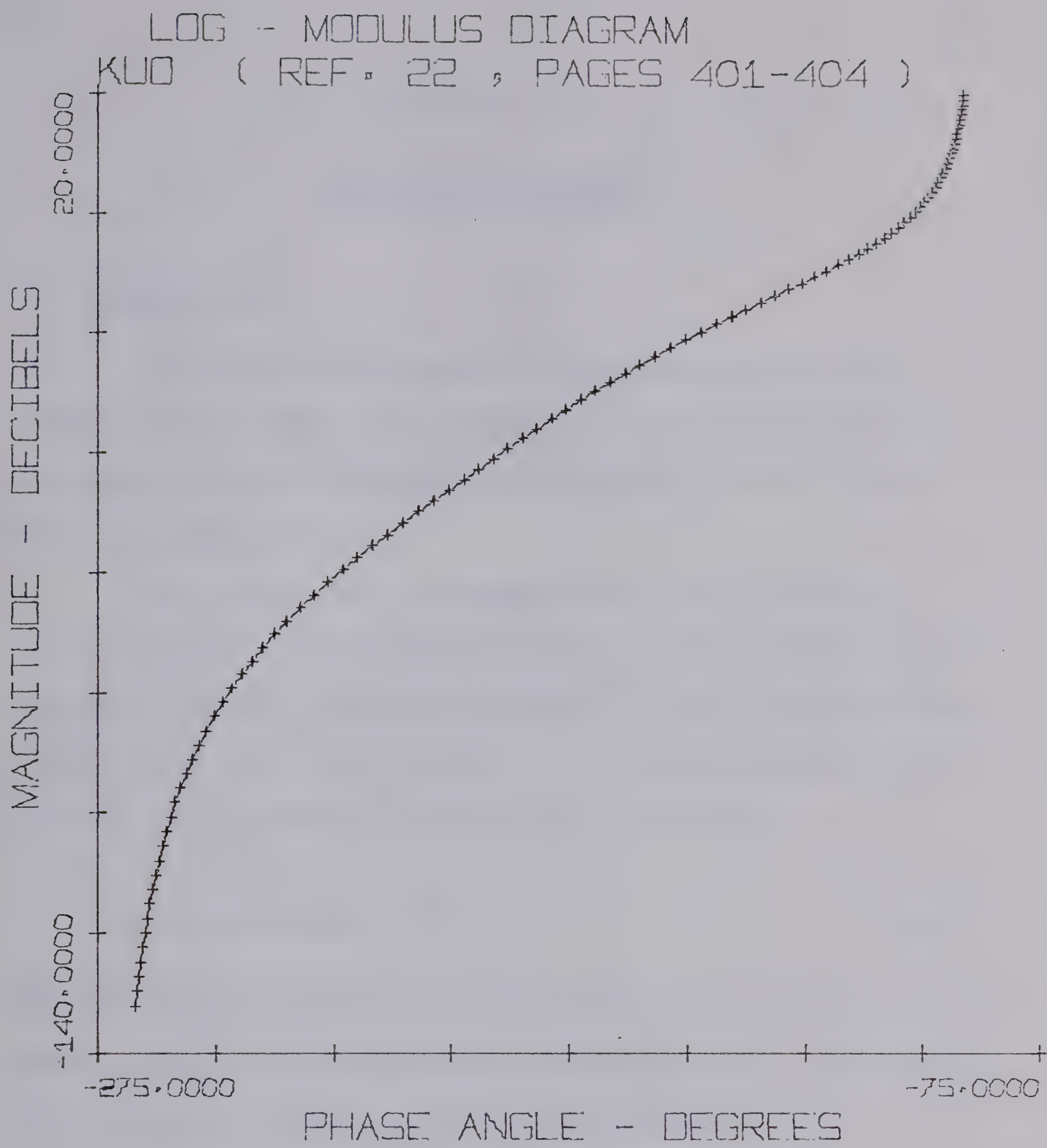


FIGURE 5-5 Second Bode Diagram



PERFORMANCE CRITERIA FOR ABOVE DIAGRAM

PHASE CROSSOVER	=	15.8116	RAD./SEC.
GAIN MARGIN	=	-34.8075	DECIBELS
GAIN CROSSOVER	=	0.9811	RAD./SEC.
PHASE MARGIN	=	77.7747	DEGREES
PEAK RESONANCE	=	0.9972	DECIBELS
RESONANT FREQUENCY	=	0.1000	RAD./SEC.

FIGURE 5-6 Second Log-Modulus Diagram

CHAPTER VI

ROOT LOCUS ANALYSIS

A. Introduction

The root locus diagram was introduced by W.R. Evans (14) in 1948. This technique has been improved and applied to the analysis and design of control systems in recent years.

The root locus is defined to be the trajectory in the s-plane followed by the poles of the closed-loop transfer function as some parameter of the corresponding system is varied continuously. The characteristic equation of any system can be written in the form

$$1 + KG(s)\,H(s) = 0 \qquad (6\text{-}1)$$

The function G(s) H(s) in general may be a rational polynomial and/or a transcendental function of s. The graphical methods of Evans based on the angle condition

$$\underline{/G(s)\,H(s)} = \pi + 2n\pi,\ n = 0, \pm, \pm\, 2, \ldots \qquad (6\text{-}2)$$

has been well developed, but it is a trial and error procedure. It can therefore be tedious and inaccurate and in such cases, an analytical solution is preferable.

B. Mathematical Basis of Calculation Methods

The open loop transfer function G(s) H(s) in

Equation 6-1 can be written as a ratio of polynomials N(s) and D(s) of degrees m and n, respectively, where m < n and K is a parameter which is proportional to the system gain

$$G(s)\ H(s) = \frac{N(s)}{D(s)}$$

$$= \frac{s^m + a_{m-1}\ s^{m-1} + a_{m-2}\ s^{m-2} + \ldots + a_o}{s^n + b_{n-1}\ s^n + b_{n-2}\ s^{n-2} + \ldots + b_o} \tag{6-3}$$

The complete frequency $\sigma + j\omega$ can now be substituted for the variable s, and by separating the real and imaginary parts of $G(\sigma + j\omega)\ H(\sigma + j\omega)$, the equations governing the root locus are given by

$$\text{Re}\ G(\sigma + j\omega)\ H(\sigma + j\omega) = -\frac{1}{k} \tag{6-4}$$

$$\text{Im}\ G(\sigma + j\omega)\ H(\sigma + j\omega) = 0 \tag{6-5}$$

It was stated in Equation 6-3 that G(s) H(s) could be given as the ratio of two polynomials N(s)/D(s), this substitution can now be made into Equations 6-4 and 6-5.

$$\frac{\text{Re}\ N(\sigma + j\omega)\ \text{Re}\ D(\sigma + j\omega) + \text{Im}\ N(\sigma + j\omega)\ \text{Im}\ D(\sigma + j\omega)}{[\text{Re}\ D(\sigma + j\omega)]^2 + [\text{Im}\ D(\sigma + j\omega)]^2} = -\frac{1}{K} \tag{6-6}$$

$$\text{Re}\ D(\sigma + j\omega)\ \text{Im}\ N(\sigma + j\omega) - \text{Re}\ N(\sigma + j\omega)\ \text{Im}\ D(\sigma + j\omega) = 0 \tag{6-7}$$

Equation 6-6 is called the "gain equation" and Equation 6-7 is referred to as the "root locus equation". If a pole of the open loop transfer function, $\sigma_1 + j\omega$, satisfies Equation 6-7, then the value of the gain corresponding to this pole can be obtained from Equation 6-6.

If N(s) and D(s) are analytic, say a polynomial in s (with real coefficients), they can be expanded into a Taylor series about any point $(\sigma, j\omega)$:

$$N(\sigma + j\omega) = N(\sigma) + j\omega N^{(1)}(\sigma) - \frac{\omega^2}{2!} N^{(2)}(\sigma) - \frac{j\omega^3}{3!} N^{(3)}(\sigma) + \ldots$$

$$D(\sigma + j\omega) = N(\sigma) + j\omega D^{(1)}(\sigma) - \frac{\omega^2}{2!} D^{(2)}(\sigma) - \frac{j\omega^3}{3!} D^{(3)}(\sigma) + \ldots \tag{6-8}$$

where $N^{(i)}$ = the ith derivative with respect to σ.

Equation 6-8 can be substituted into the Equations 6-6 and 6-7 and after some simplifications new expressions can be obtained for describing the root locus.

The root locus equation can be expressed as

$$[R_1(\sigma) - \omega^2 R_3(\sigma) + \omega^4 R_5(\sigma) \pm \ldots] = 0 \tag{6-9}$$

while the gain equation is given by

$$\frac{R_o(\sigma) - \omega^2 R_2(\sigma) + \omega^4 R_4(\sigma) - \omega^6 R_6(\sigma) \pm \ldots}{D_o(\sigma) - \omega^2 D_2(\sigma) + \omega^4 D_4(\sigma) - \omega^6 D_6(\sigma) \pm \ldots} = -\frac{1}{k} \tag{6-10}$$

The coefficients of the powers of ω^2 in the previous two equations are given by

$$R_k(\sigma) = \sum_{r=o}^{k} (-1)^r \frac{N^{(r)}(\sigma)}{r!} \frac{D^{(k-r)}(\sigma)}{(k-r)!} \qquad (6\text{-}11)$$

$$D_k(\sigma) = \sum_{r=o}^{k} (-1)^r \frac{D^{(r)}(\sigma)}{r!} \frac{D^{(k-r)}(\sigma)}{(k-r)!} \qquad (6\text{-}12)$$

The degrees of Equation 6-11 and 6-12 are given by $(n + m - k)$ and $(2n - k)$, respectively.

A number of observations can be made concerning the root locus equation.

a) The real axis $\omega = 0$ is always on the root locus.

b) The locus off the axis is expressed by a finite series in ω^2.

c) The last term in Equation 6-9 is given by

$(-1)^{(n+m-1)/2}\ \omega^{n+m-1}\ R_{n+m}(\sigma)$ if n+m is odd, and

$(-1)^{(n+m-2)/2}\ \omega^{n+m-2}\ R_{n+m-1}(\sigma)$ if n+m is even.

d) The root locus equation represents an algebraic curve and is derived for a given ratio of polynomials without any prior knowledge of the location of the poles and zeros of the open loop transfer function.

e) Terms of powers higher than ω^4 are present in Equation (6-9) only if the sum of the degrees of the numerator and denominator of the open loop transfer function exceeeds 6.

f) The breakaway points on the real axis are the intersections of the root locus and the real

axis and are given by the solutions to the equation

$$R_1(\sigma) = 0$$

The coefficients of the powers of ω^2 in the Taylor series expansion of $N(s)$ and $D(s)$ given by Equation 6-8 are first arranged in an array called the root locus array.

$$\begin{array}{cccccccc}
R_0(\sigma) & R_1(\sigma) & R_2(\sigma) & R_3(\sigma) & & R_m(\sigma) & & R_n(\sigma) \\
N(\sigma) & -N^{(1)}(\sigma) & \dfrac{N^{(2)}(\sigma)}{2!} & -\dfrac{N^{(3)}(\sigma)}{3!} & \cdots(-1)^m & \dfrac{N^{(m)}(\sigma)}{m!} & \cdots & 0 \\
D(\sigma) & D^{(1)}(\sigma) & \dfrac{D^{(2)}(\sigma)}{2!} & \dfrac{D^{(3)}(\sigma)}{3!} & \cdots & \dfrac{D^{(m)}(\sigma)}{m!} & \cdots & \dfrac{D^{(n)}(\sigma)}{n!}
\end{array}$$

In this array the odd order differentiations of the numerator $N(\tau)$ are negative. The coefficients of $R_k(\sigma)$ are obtained by taking the sum of the $(k + 1)$ cross products. For example the calculation of $R_2(\sigma)$ would be as follows:

$$R_2(\sigma) = \begin{bmatrix} N(\sigma) & -N^{(1)}(\sigma) & \dfrac{N^{(2)}(\sigma)}{2!} \\ D(\sigma) & D'(\sigma) & \dfrac{D^{(2)}(\sigma)}{2!} \end{bmatrix}$$

$$= N(\sigma)\,\frac{D^{(2)}(\sigma)}{2!} - N^{(1)}(\sigma)\,D'(\sigma) + \frac{N^{(2)}(\sigma)}{2!}\,D(\sigma)$$

The coefficients $D_0(\sigma)$, $D_2(\sigma)$, $D_4(\sigma)$, etc. are obtained in an analogous manner.

C. Computer Implementation of Root Locus Algorithm

The root locus algorithm that is described in the previous section and summarized in the form of Equation (6-9) and (6-10) was implemented in a FORTRAN program. The derivatives m and n derivatives of the numerator and denominator polynomials, respectively, are first calculated. This enables the root locus array to be calculated. Subroutine RCOEF determines the $R_k(\sigma)$ coefficients by Equation (6-11) while subroutine DCOEF and Equation (6-12) are used to calculate the $D_k(\sigma)$ coefficients.

The root locus equation is then evaluated by subroutine RLEVL to determine all the values of the frequency that satisfy the equation. The real axis is divided into fifty segments and its length is governed by the low and high frequency limits that the user was requested by subroutine RLINP to enter. Successively these fifty values are substituted into the root locus equation and the solutions that give a positive value of the imaginary frequency component are determined. The frequency with a zero imaginary component is always a solution.

The values of the gain associated with each of the frequencies determined by RLEVL is now calculated from the gain Equation (6-10) by the subroutine GAIN. The root locus algorithm that has been used results in frequencies that give both positive and negative values of the gain parameter K which implies negative and positive feedback respectively.

D. Display of the Root Locus Diagram

The root locus data is displayed onto a point-plot by subroutine RLPLT. Both the positive and negative values of the gain are placed onto this plot. They are differentiated by plotting the frequencies points with positive gain as small plus signs (+) while those points with negative gain are plotted as small crosses (x). The gain values are not plotted onto the graph but can be obtained from the optional printout of the values used to plot the root locus. Three examples of the root locus diagrams obtained are given here as

i) $G(s)\ H(s) \quad \dfrac{K}{(s + 1)\ (s + 2)\ (s + 3)}$ Figure 6-1

ii) $G(s)\ H(s) \quad \dfrac{K}{s^4 + 16s^3 + 108s^2 + 400s + 800}$

Figure 6-2

iii) $G(s)\ H(s) \quad \dfrac{K}{s(s + 1.8)(s^2 + 2s + 2)}$ Figure 6-3

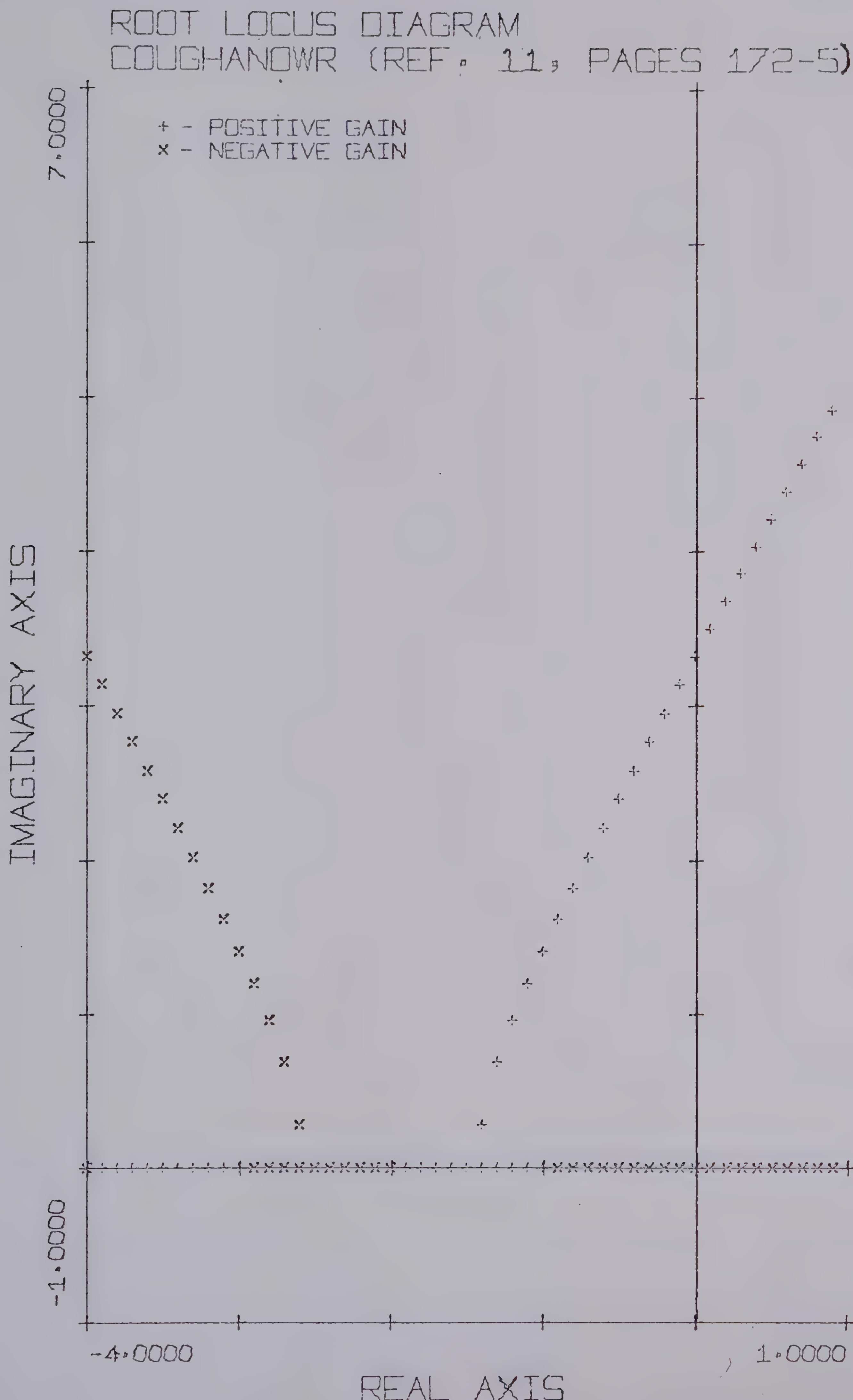


FIGURE 6-1 First Example of Root Locus Diagram

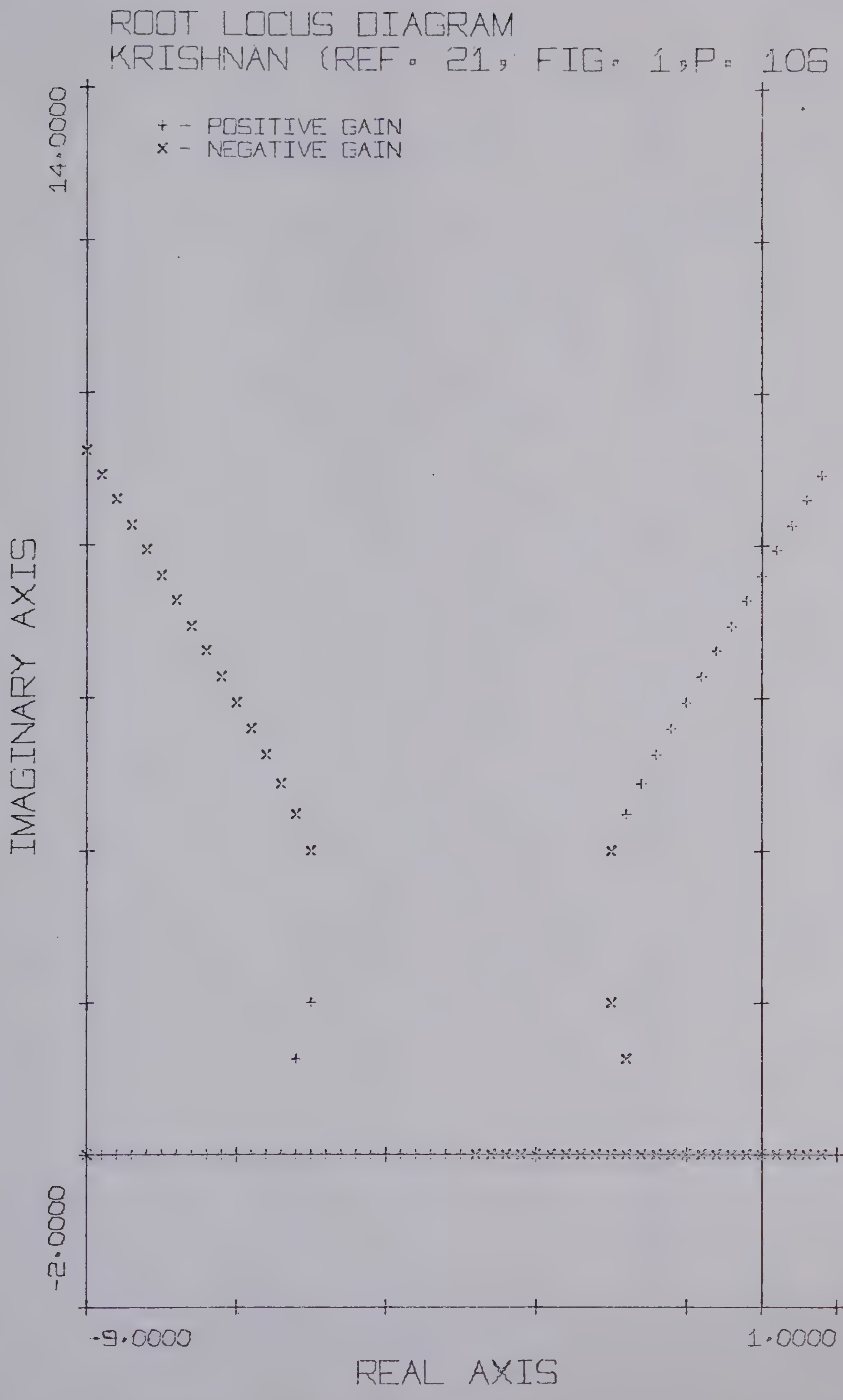


FIGURE 6-2 Second Example of Root Locus Diagram

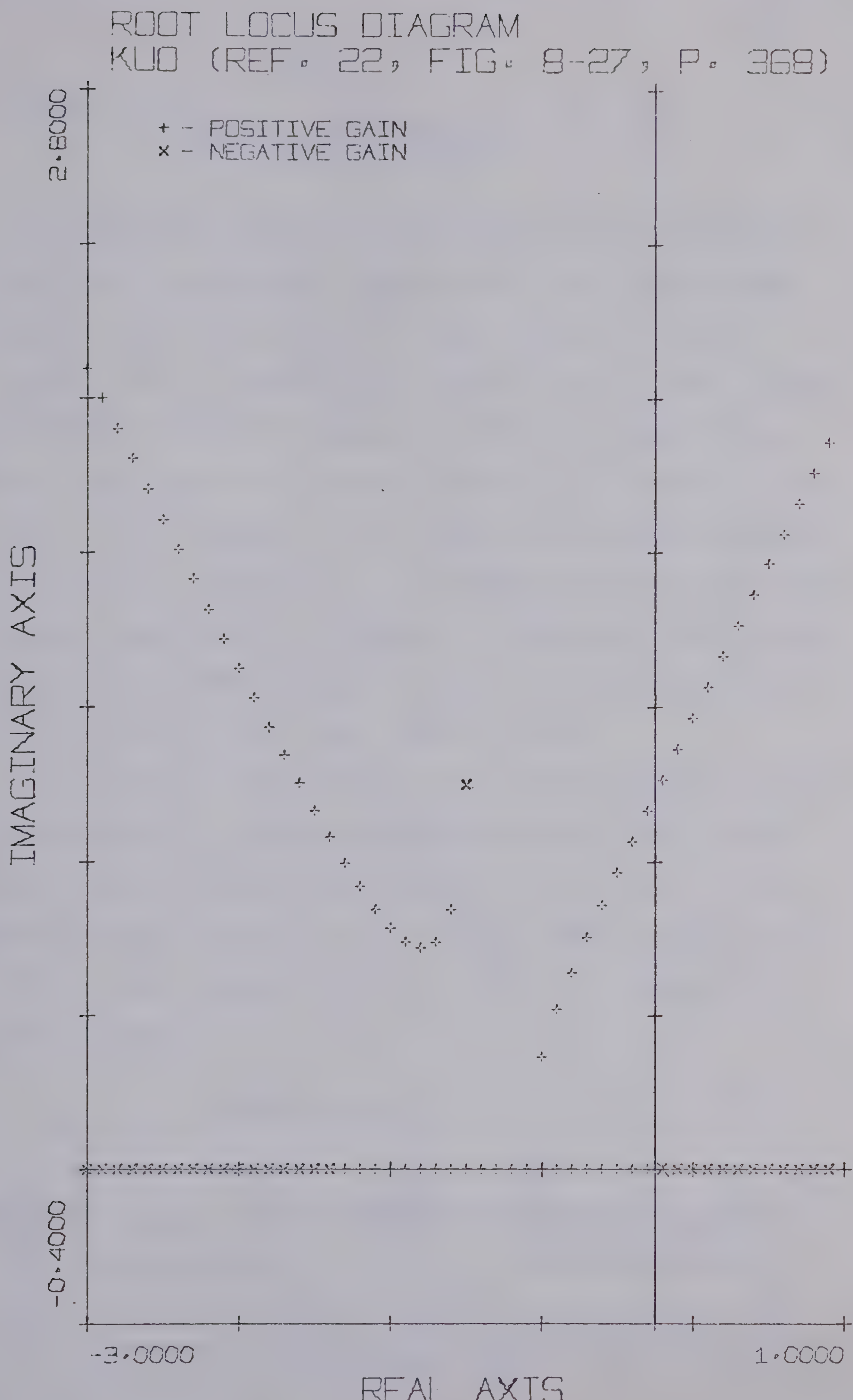


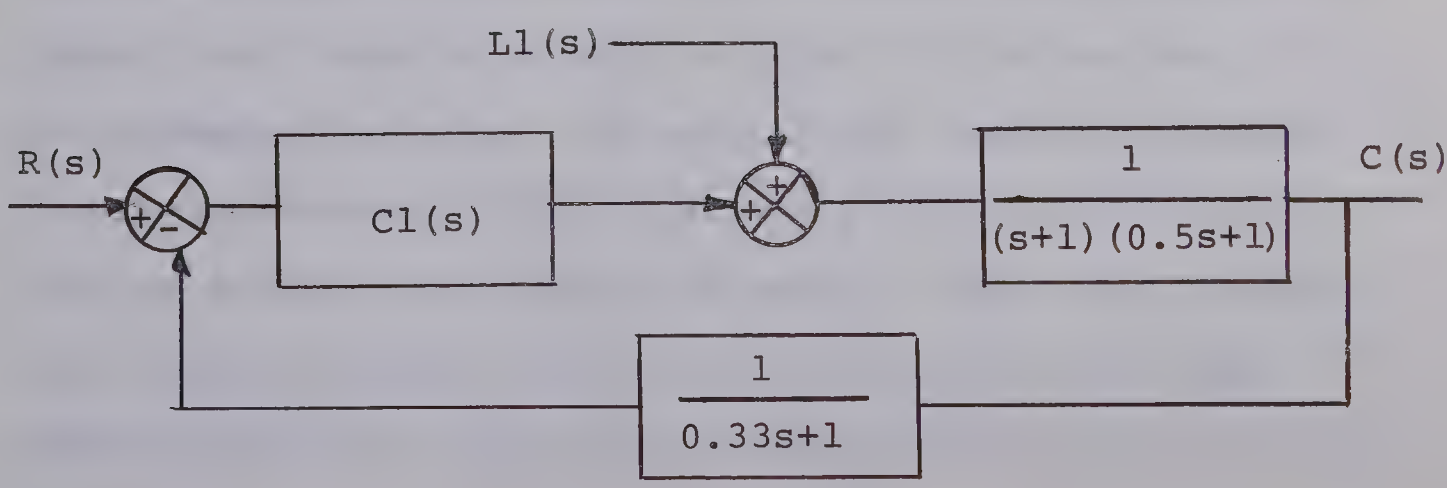
FIGURE 6-3 Third Example of Root Locus Diagram

CHAPTER VII

PROGRAM DEMONSTRATION

The preceding chapters have presented the mathematical basis for the different computations, and given example plots and a brief description of each section of CSDAP. This chapter will illustrate one typical use of CSDAP, namely a study of the effect of changing controller types and constants on the output of a control system. The various plots are given in this chapter, while computer generated documentation is given in Appendix B. Detailed operating procedures and further examples of all possible information that can be obtained from the CSDAP system is given in Appendix A which has been written to serve as a User's Manual.

The control system studied is a single feedback control loop consisting of a process described by two first order transfer functions with time constants of 1.0 and 0.5, and a feedback element described as a first order transfer function with a time constant of 0.33. The block diagram is illustrated below.



This control loop will be used to illustrate how the CSDAP system can be used to study the effect of different controllers on the response of the system C(t) when a unit step input is applied to the system as a load disturbance, (L1(t)). The results will be expressed by means of the transient response, the frequency response (Bode, Nyquist and Log Modulus), and the root locus diagrams.

The CSDAP system is initiated as a non-process program from the card reader. A listing of the control digit and data switch functions is obtained on the typewriter by previously setting data switch zero in the "ON" position (see section A.2.2. of Appendix A). The first function that must be performed is the configuration of the control system. This is accomplished by entering a CONTROL DIGIT of 1 and then the correct transfer functions. Data switch 2 is set to the "ON" position before entering the CONTROL DIGIT of 1 and causes a listing of the transfer functions to be made after they are correctly entered into the system. Since, for the initial configuration, a proportional controller with a gain of 1 is desired, it is not necessary to enter the control one transfer function because any transfer function that is not entered by the user is assumed to be equal to unity. After the process and feedback element transfer functions have both been entered the user terminates this section of the program by

use of the END-OF-FIELD (EOF) character. This results in the statement "ILLEGAL TRANSFER FUNCTION DATA" being typed out to the user and causes the program to request the next CONTROL DIGIT.

The user now instructs the CSDAP system to calculate the closed loop transfer function and the inverse Laplace transformation for the system response desired. Data switch 2 is reset to the "OFF" position and data switch 3 is set to the "ON" position to give the input/output variable and forcing function selection options. A CONTROL DIGIT of 2 is then entered. The system response C(t) to a unit step in the load variable (L1) is desired so the set of variables shown on page B-2 is entered. Data switch 3 is then reset to the "OFF" position.

Since the user has a CSDAP User's Manual (Appendix A) to consult for variable options he resets all the data switches to the "OFF" position to restrict further assistance from the program. The user then proceeds to have CSDAP calculate the transient response of the configurated control system by entering 3 for the requested CONTROL DIGIT. A zero is entered for the time response replot digit since the transient response data has not been previously calculated. On the basis of information provided by the user, 250 transient response values over a time range from 0-25 seconds are calculated. The data is then plotted onto the storage oscilloscope as an upper-half plot with no title. The

performance criteria calculated from these transient response values are then displayed on the bottom half of this oscilloscope by entering a CONTROL DIGIT of 4 and an output device option of 1. After reviewing the oscilloscope display a plot of the transient response and performance criteria was obtained on the digital plotter by specifying a plot device of 0 instead of 1 in the previous parameter specifications that caused the display to be put on the oscilloscope.

The next step in the study was to vary the proportional gain constant to satisfy the arbitrarily chosen criteria of about 30 percent overshoot. The proportional gain was changed by entering a new transfer function for controller one, C1(s). The gain was set at 0.5 and the transient response calculated and displayed on the oscilloscope and the performance criteria printed on the typewriter. The copy of the transient response curve that was plotted from the data calculated for the oscilloscope display is shown in the upper-half of Figure 7-2. The controller gain was then set to 0.75 and the transient response curve that resulted is shown as the bottom-half of Figure 7-2. The performance criteria were then displayed on the oscilloscope. It was necessary for the user to manually erase the oscilloscope before requesting the performance criteria to be displayed because normally the criteria are displayed below an upper-half page plot and the oscilloscope would have been erased for that plot.

PERFORMANCE CRITERIA FOR ABOVE RESPONSE

STEADY STATE	=	0.7500
PERCENT OVERSHOOT	=	47.18
PEAK TIME	=	1.2048
RISE TIME 0 TO 100	=	0.6370
RISE TIME 10 TO 90	=	0.3529
DELAY TIME	=	0.3420
DECAY RATIO	=	0.1617
SETTLING TIME	=	4.5180

ERROR INTEGRALS

IAE	=	0.79296
ISE	=	0.24675
ITAE	=	1.27598

Figure 7-1. Unity Gain Proportional Control

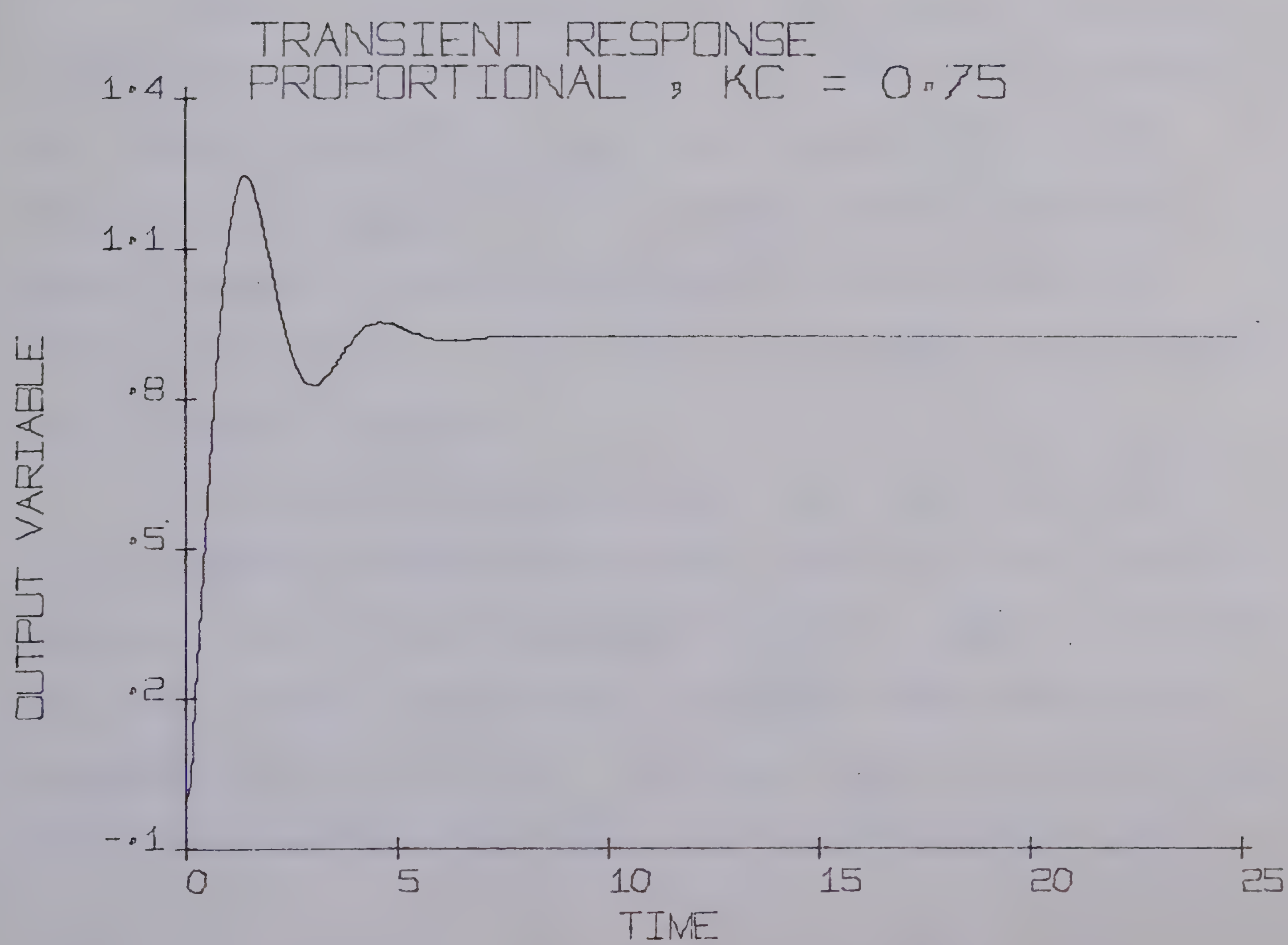


Figure 7-2. Proportional Control with Variable Gains

The displayed performance criteria indicated an overshoot of about 35 percent. On the basis of the results a value of the gain constant equal to 0.65 was chosen and a plot of the transient response and the list of associated performance criteria obtained on the digital plotter. Before entering the CONTROL DIGIT 3, data switch 6 was set to the "ON" position to obtain the tabulation of the transient response data on the line printer. These values are given on page B-22 of Appendix B and are the same as those used for the plot shown in Figure 7-3. After the listing had been completed the data switch was reset to the "OFF" position.

The user then decided to list the overall transfer function for the existing system. This was done by entering a CONTROL DIGIT of 7 and a list device option of 0 indicating the typewriter. A transfer function name of TM was entered to obtain the overall transfer function listing (page A-23). The resulting transfer function listing is shown on page B-7.

The controller gain constant was then changed back to unity and the frequency response and root locus diagrams obtained. The use of logarithmic scales in addition to the unity gain offers substantial advantages when making calculations about the system. The gain margin can be obtained direct from the Bode diagram since it is the difference

PERFORMANCE CRITERIA FOR ABOVE RESPONSE

STEADY STATE	=	1.0169
PERCENT OVERSHOOT	=	29.64
PEAK TIME	=	1.5531
RISE TIME 0 TO 100	=	0.9012
RISE TIME 10 TO 90	=	0.6518
DELAY TIME	=	0.4886
DECAY RATIO	=	0.0668
SETTLING TIME	=	3.7074

ERROR INTEGRALS

IAE	=	0.91439
ISE	=	0.43941
ITAE	=	1.09422

Figure 7-3. Proportional Control Only.

between the overall gain and unity at a frequency equal to 180 degrees lag.

The frequency response data was calculated for a frequency range from 0.01 to 100.0 radians per second. After displaying the Bode diagram on the oscilloscope it was decided to set the range from 0.1 to 100.0 radians per second. This removed the first decade from the plot which was of a constant phase angle. The Nyquist and Log Modulus plots are constructed using the data previously calculated for the Bode diagram. An automatically scaled Nyquist diagram is first displayed on the oscilloscope before setting the range of values for each axis of the plot to be drawn on the digital plotter. The Bode, Nyquist and Log Modulus diagrams are shown in Figures 7-4, 7-5, and 7-6 respectively. After the desired frequency range is chosen for the root locus diagram by a similar method to that used for the Bode diagram it was plotted and shown as Figure 7-7.

The next phase of the study was to investigate the effect on the system of a proportional - integral controller when the integral time constant was varied. The transient response curves for integral times of 0.5 and 0.75 and a proportional constant of 0.65 are given in Figure 7-8. The transient response curve and the performance criteria for an integral time constant of 1.10 are shown in Figure 7-9. The value of the integral time equal to

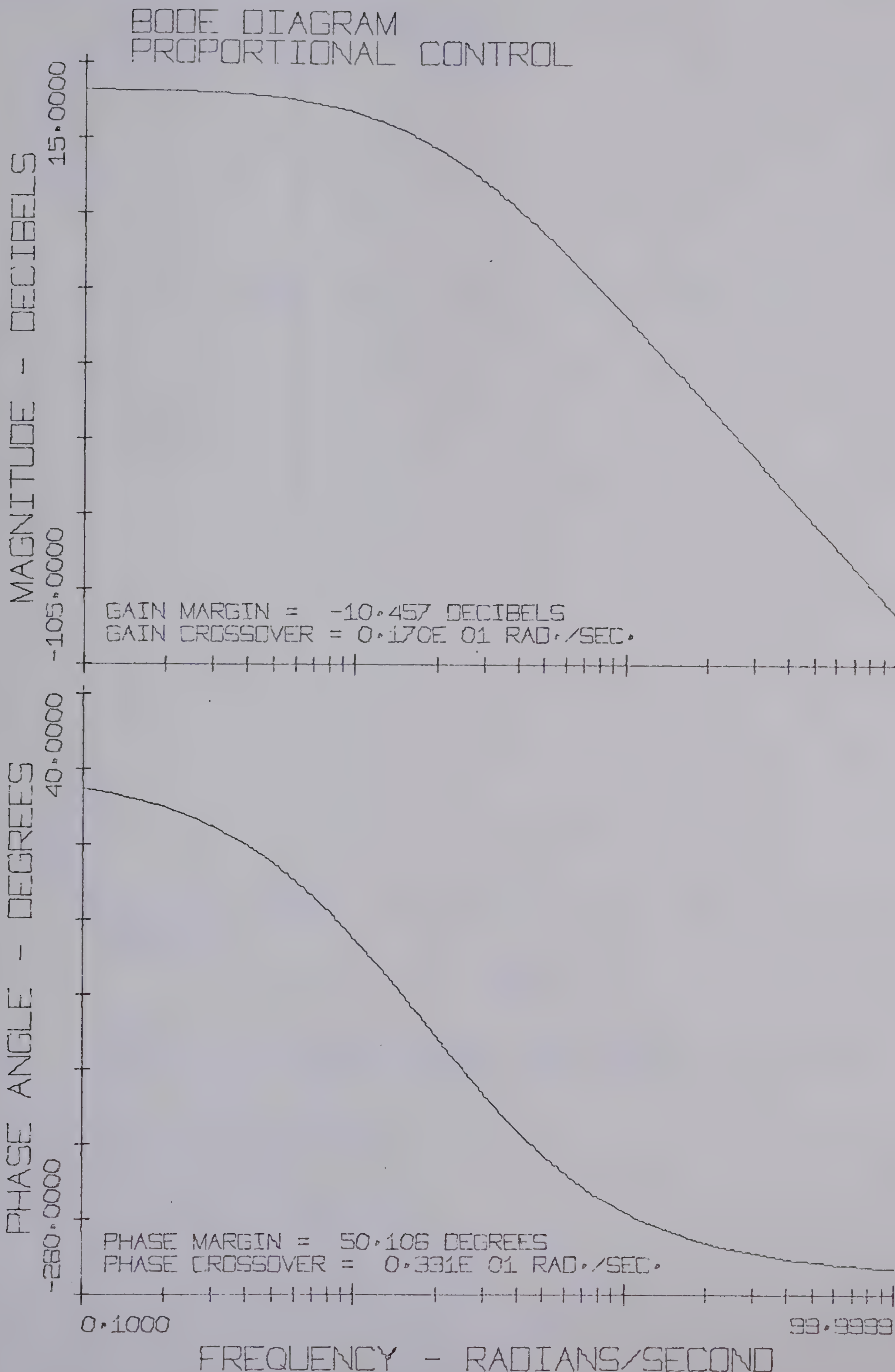


Figure 7-4. Bode Diagram for Proportional Control Only.

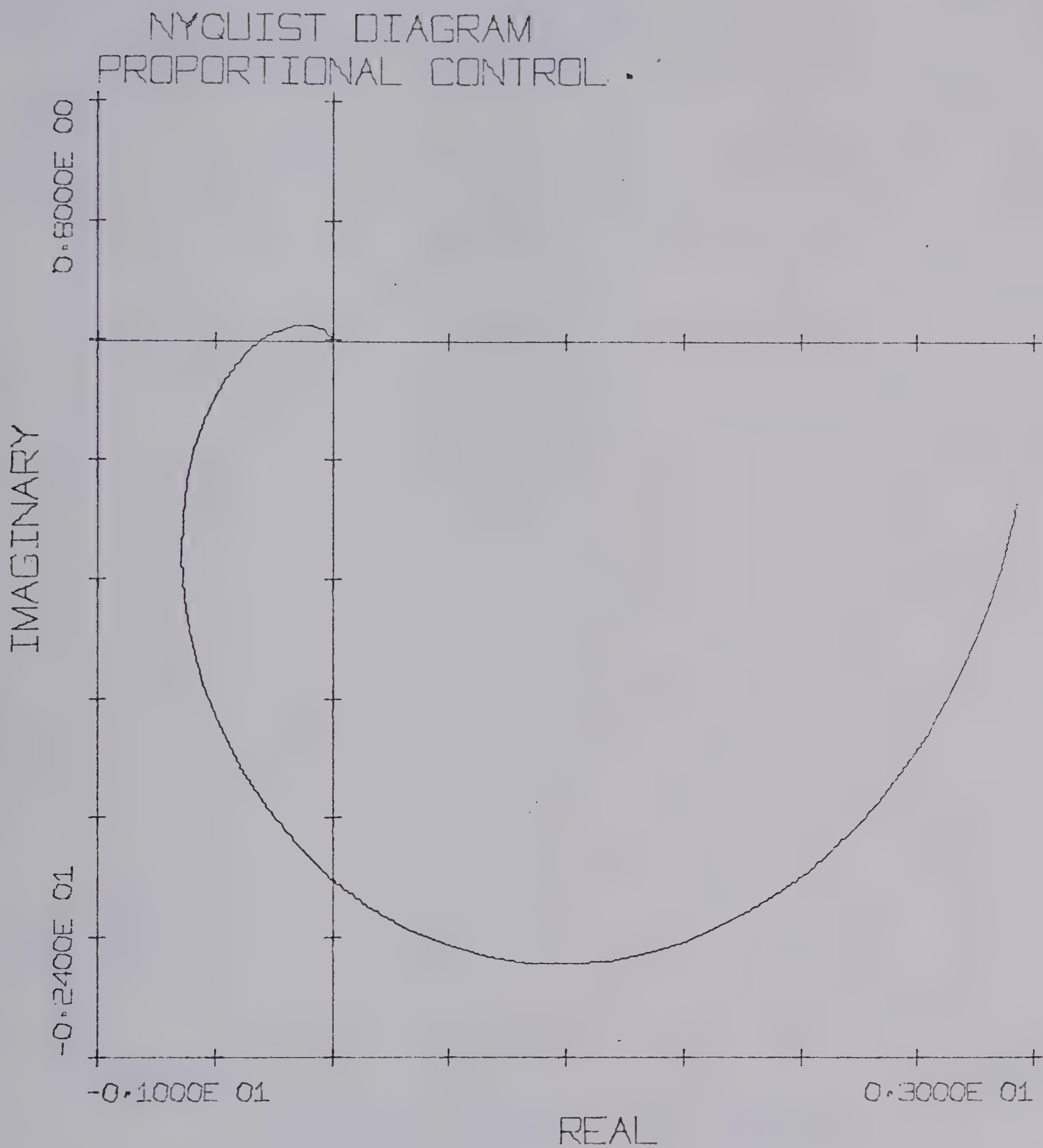
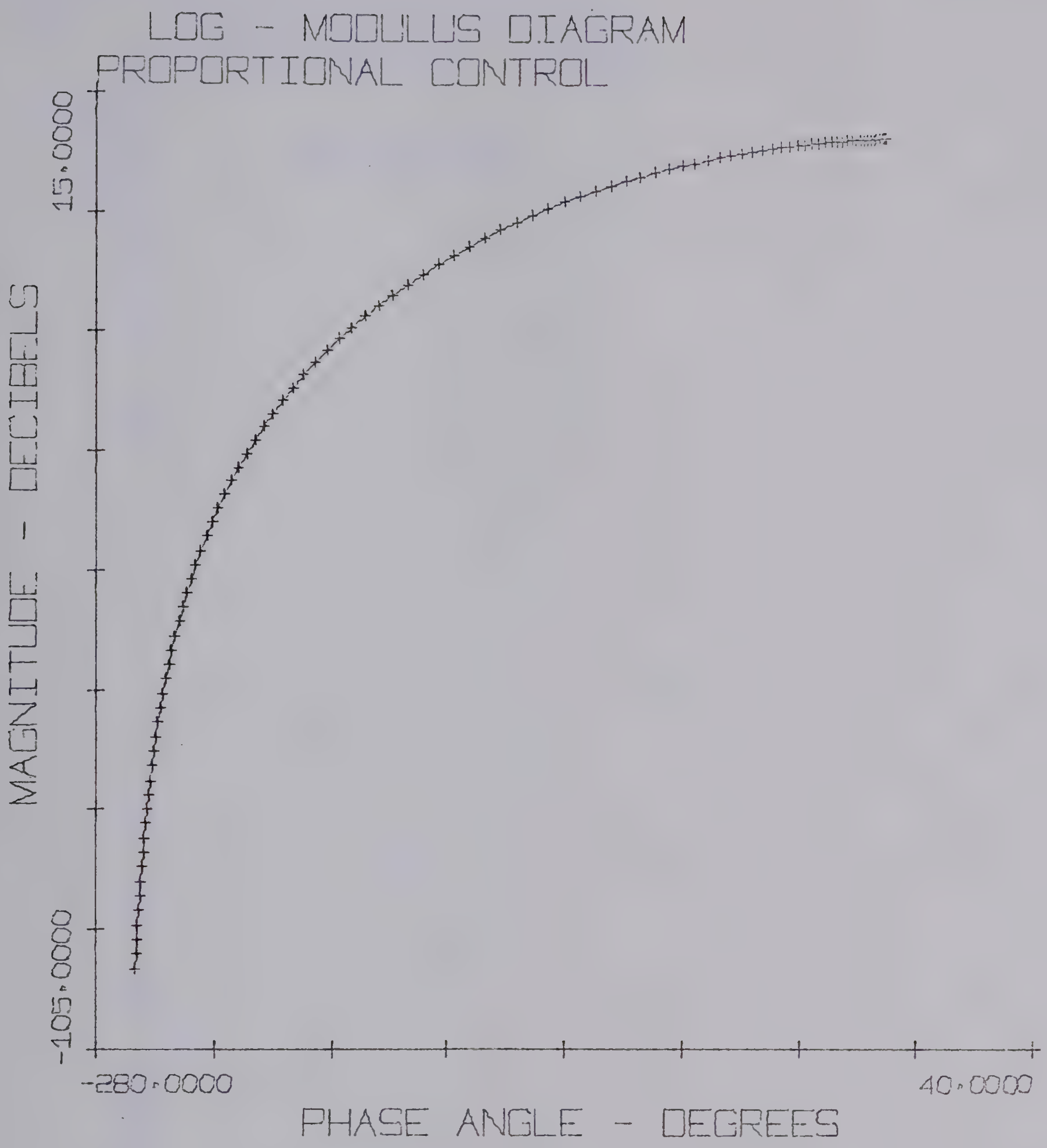


Figure 7-5. Nyquist Diagram For Proportional Control Only.



PERFORMANCE CRITERIA FOR ABOVE DIAGRAM

PHASE CROSSOVER	=	3.3165	RAD./SEC.
GAIN MARGIN	=	-10.4573	DECIBELS
GAIN CROSSOVER	=	1.7099	RAD./SEC.
PHASE MARGIN	=	50.1060	DEGREES
PEAK RESONANCE	=	1.2866	DECIBELS
RESONANT FREQUENCY	=	2.0216	RAD./SEC.

Figure 7-6. Log-Modulus Diagram for Porportional Control Only.

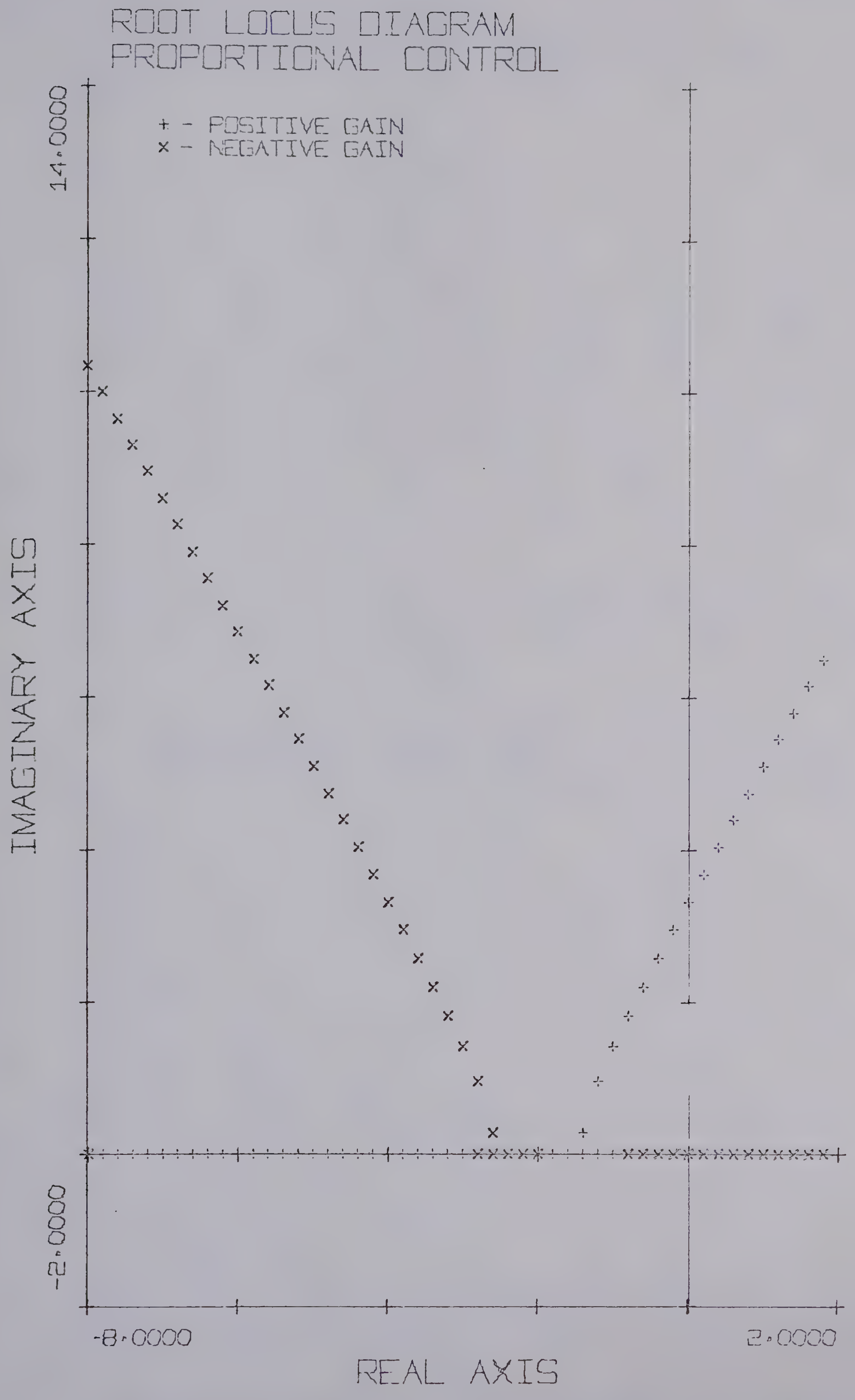


Figure 7-7. Root Locus Diagram For Proportional Control Only.

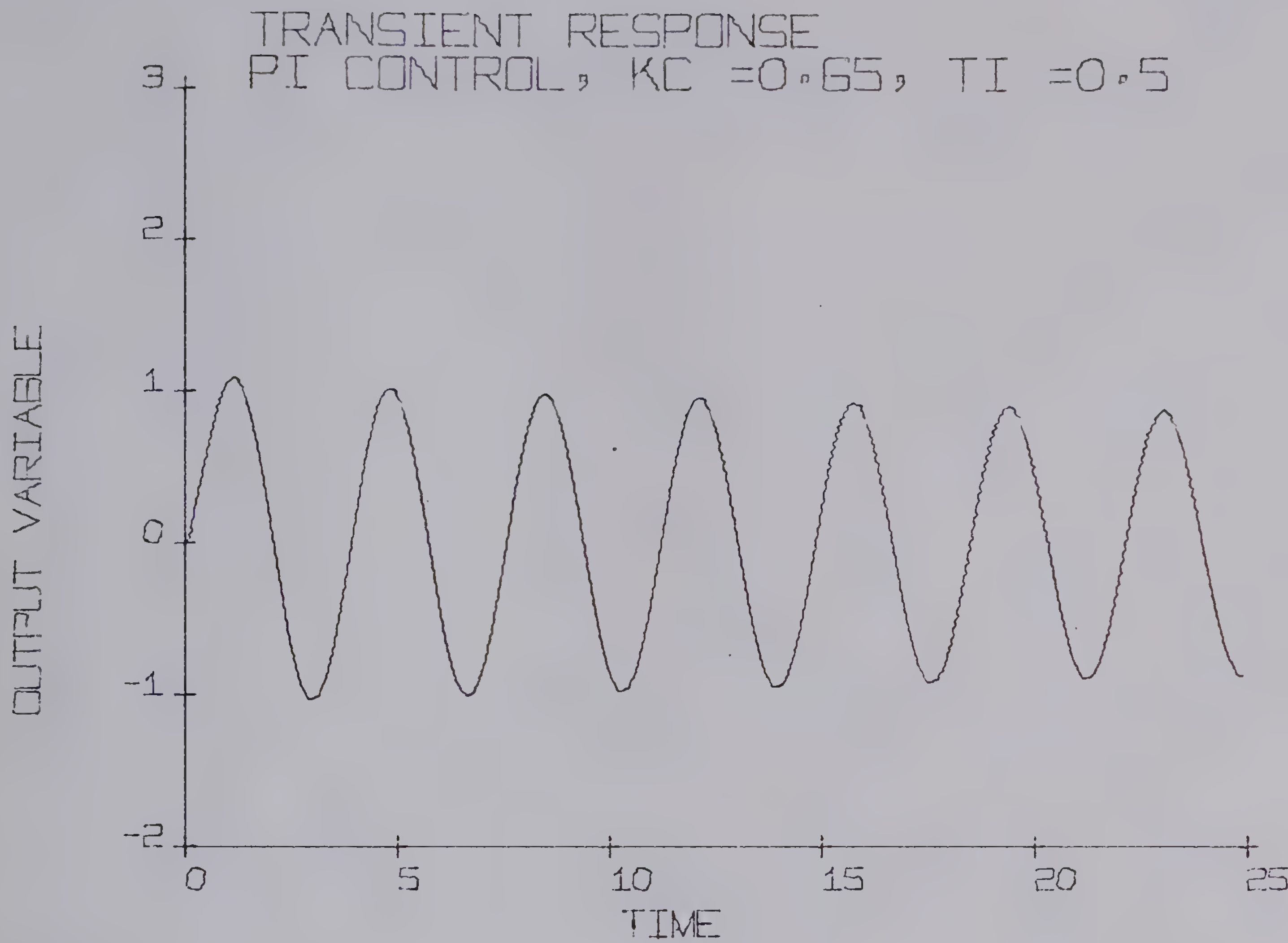
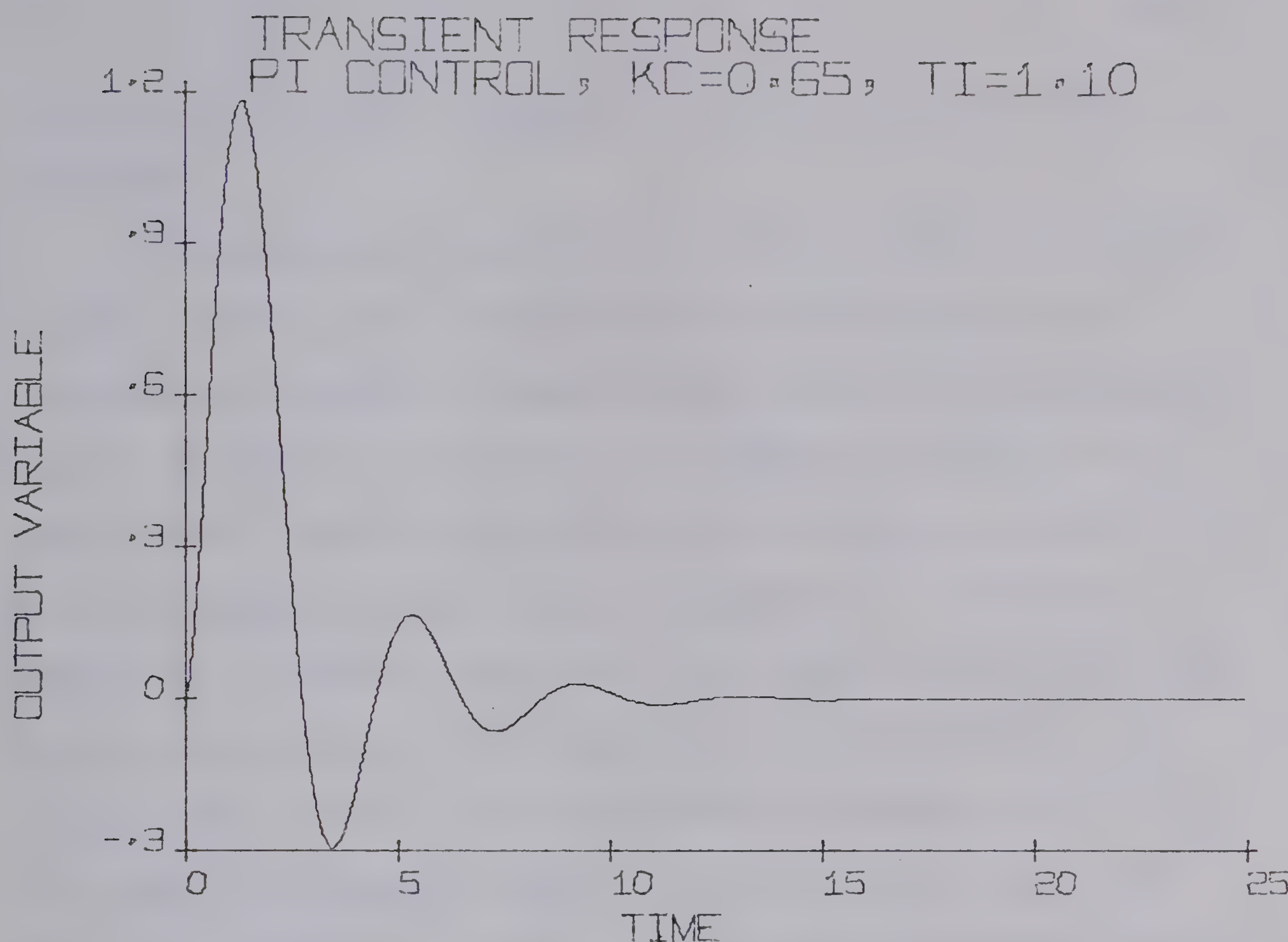


Figure 7-8. Variable Integral Action For Proportional Integral Control.



PERFORMANCE CRITERIA FOR ABOVE RESPONSE

STEADY STATE	=	0.0000
PERCENT OVERSHOOT	=	0.00
PEAK TIME	=	1.3026
RISE TIME 0 TO 100	=	0.0002
RISE TIME 10 TO 90	=	0.0001
DELAY TIME	=	0.0001
DECAY RATIO	=	0.1401
SETTLING TIME	=	13.3266

ERROR INTEGRALS

IAE	=	2.52954
ISE	=	1.78712
ITAE	=	6.05329

Figure 7-9. Proportional-Integral Control

1.10 was arbitrarily chosen as the best integral time constant.

The transfer functions were then listed. A partial listing showing only the control one, process two, and feedback one transfer functions was made on the typewriter, and it is shown on page B-13. A complete listing of all the transfer functions was obtained on the line printer as shown on pages B-23 to B-25 of Appendix B. The Bode, Nyquist, Log Modulus and Root Locus diagrams obtained from the plotter by a similar procedure to that used for the proportional control only are shown as Figures 7-10, 7-11, 7-12 and 7-13, respectively. Before the final Bode diagram was plotted data switch 10 was set in the "ON" position to provide a listing of the frequency response values. The line printer listing is given on pages B-26 and B-27 in Appendix B. The root locus values also listed to the line printer are given as pages B-28 and B-29 in Appendix B by having data switch 14 set in the "ON" position when the final root locus diagram was being plotted.

The effect of adding derivative action to the previously determined proportional integral controller was then studied. Figure 7-14 shows the transient response curves for values of the derivative constant equal to 0.5 and 0.25 while Figure 7-15 shows the transient response curve and the performance criteria for a third value of the derivative constant that was employed. For this value of

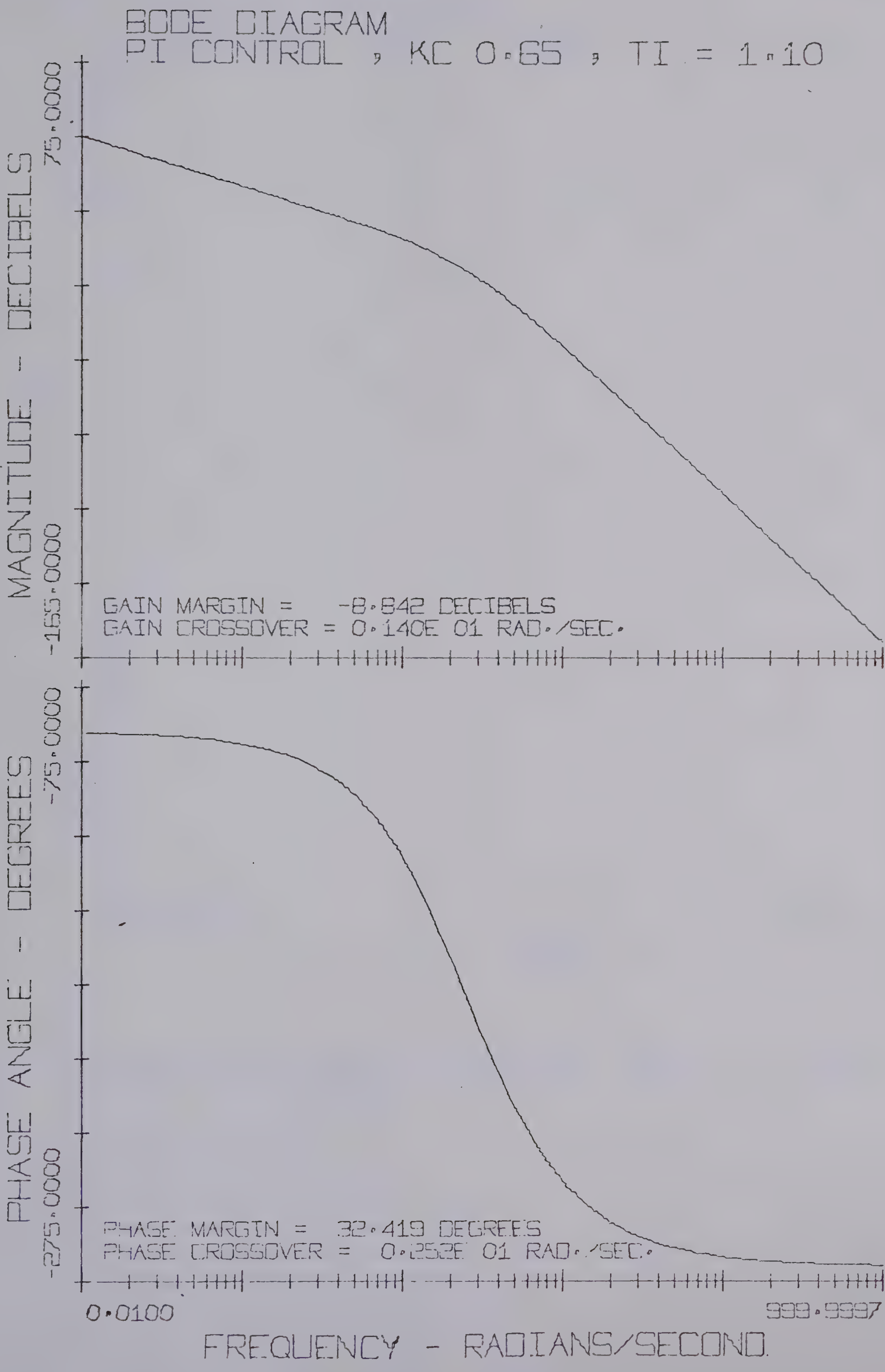
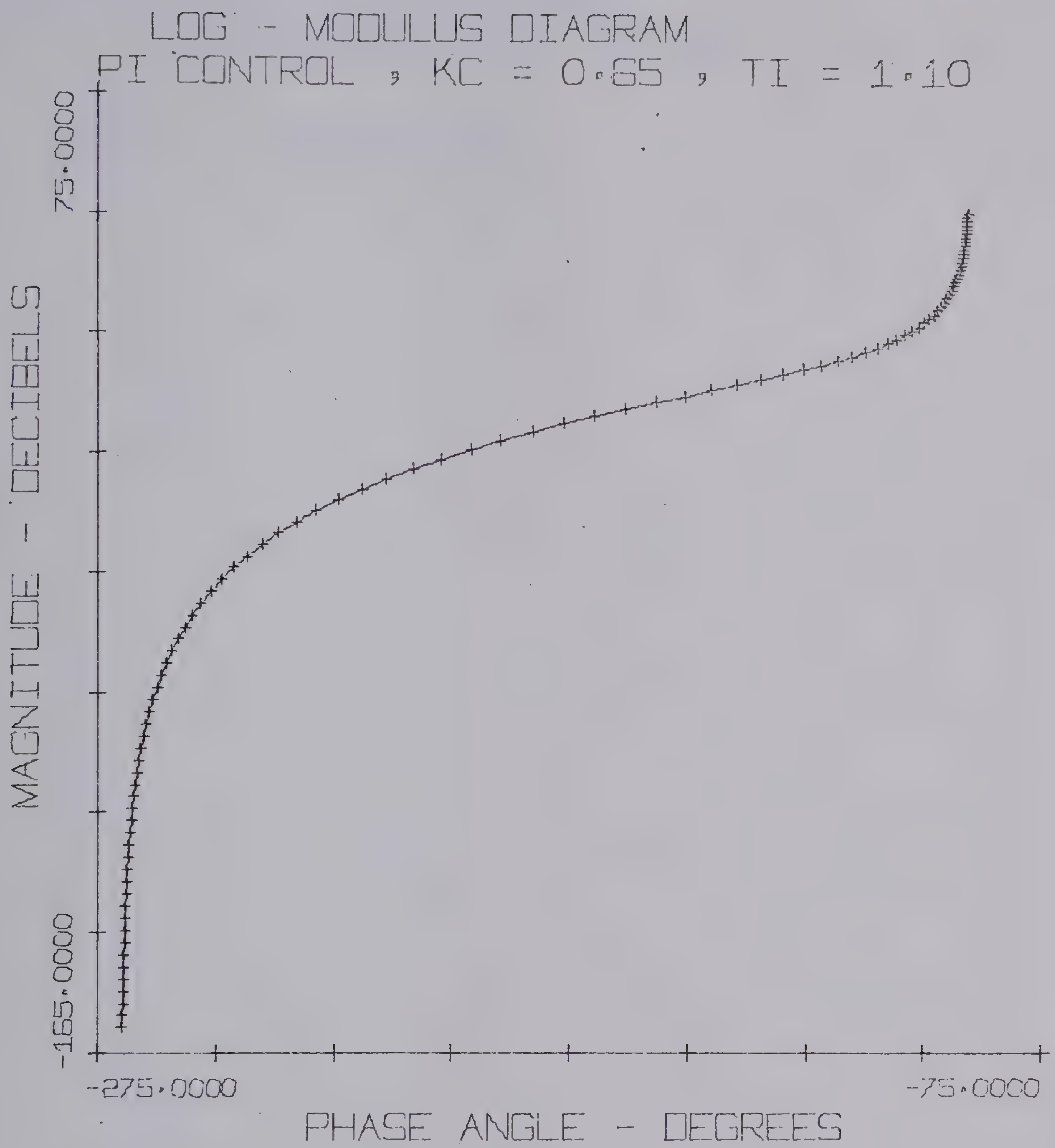


Figure 7-10. Bode Diagram for Proportional-Integral Control.

Figure 7-11. Nyquist Diagram For Proportional-Integral Control.



PERFORMANCE CRITERIA FOR ABOVE DIAGRAM

PHASE CROSSOVER	=	2.5292	RAD./SEC.
GAIN MARGIN	=	-8.8425	DECIBELS
GAIN CROSSOVER	=	1.4030	RAD./SEC.
PHASE MARGIN	=	32.4195	DEGREES
PEAK RESONANCE	=	1.8611	DECIBELS
RESONANT FREQUENCY	=	1.5308	RAD./SEC.

Figure 7-12. Log-Modulus Diagram for Proportional-Integral Control.

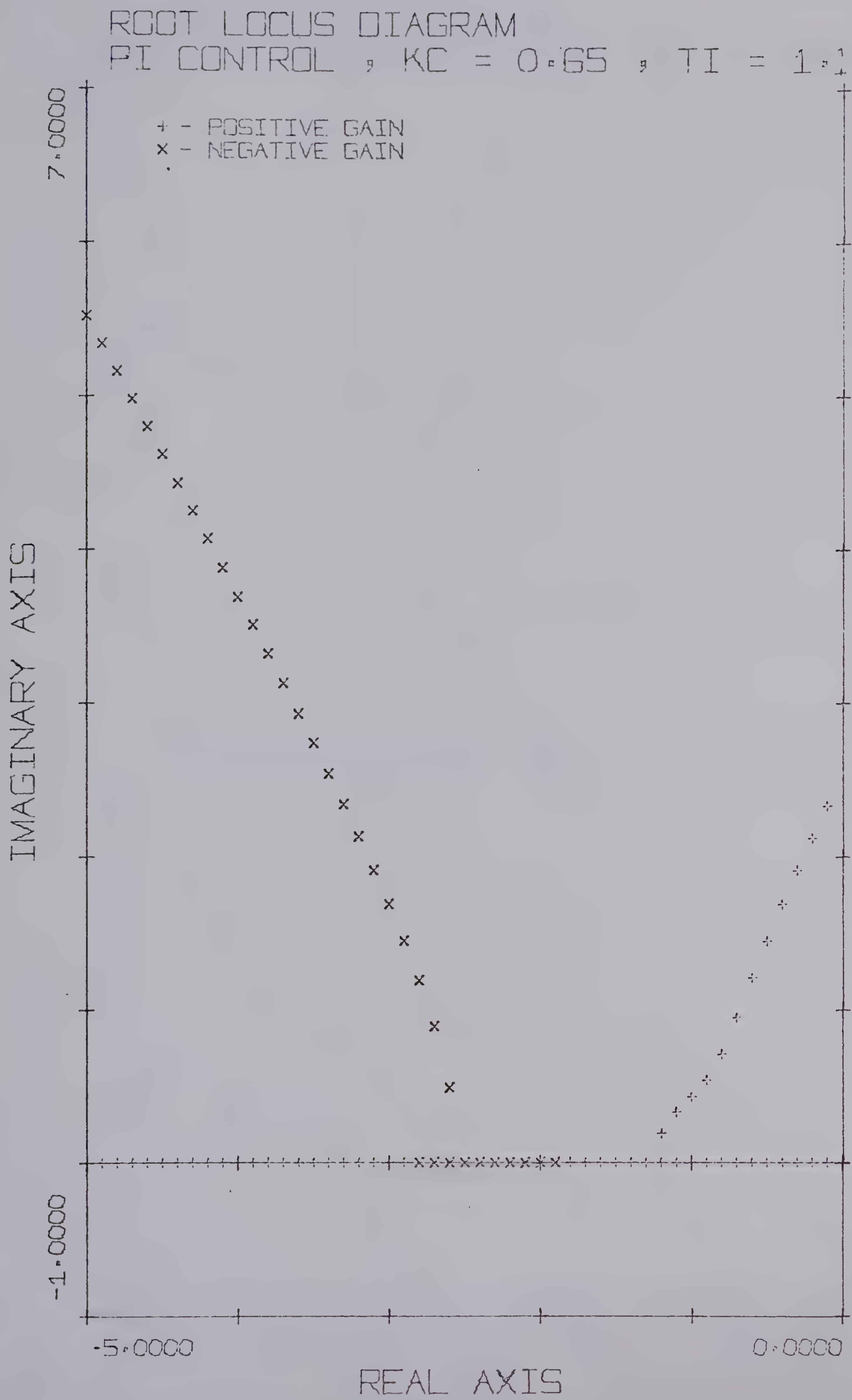


Figure 7-13. Root Locus Diagram For Proportional-Integral Control.

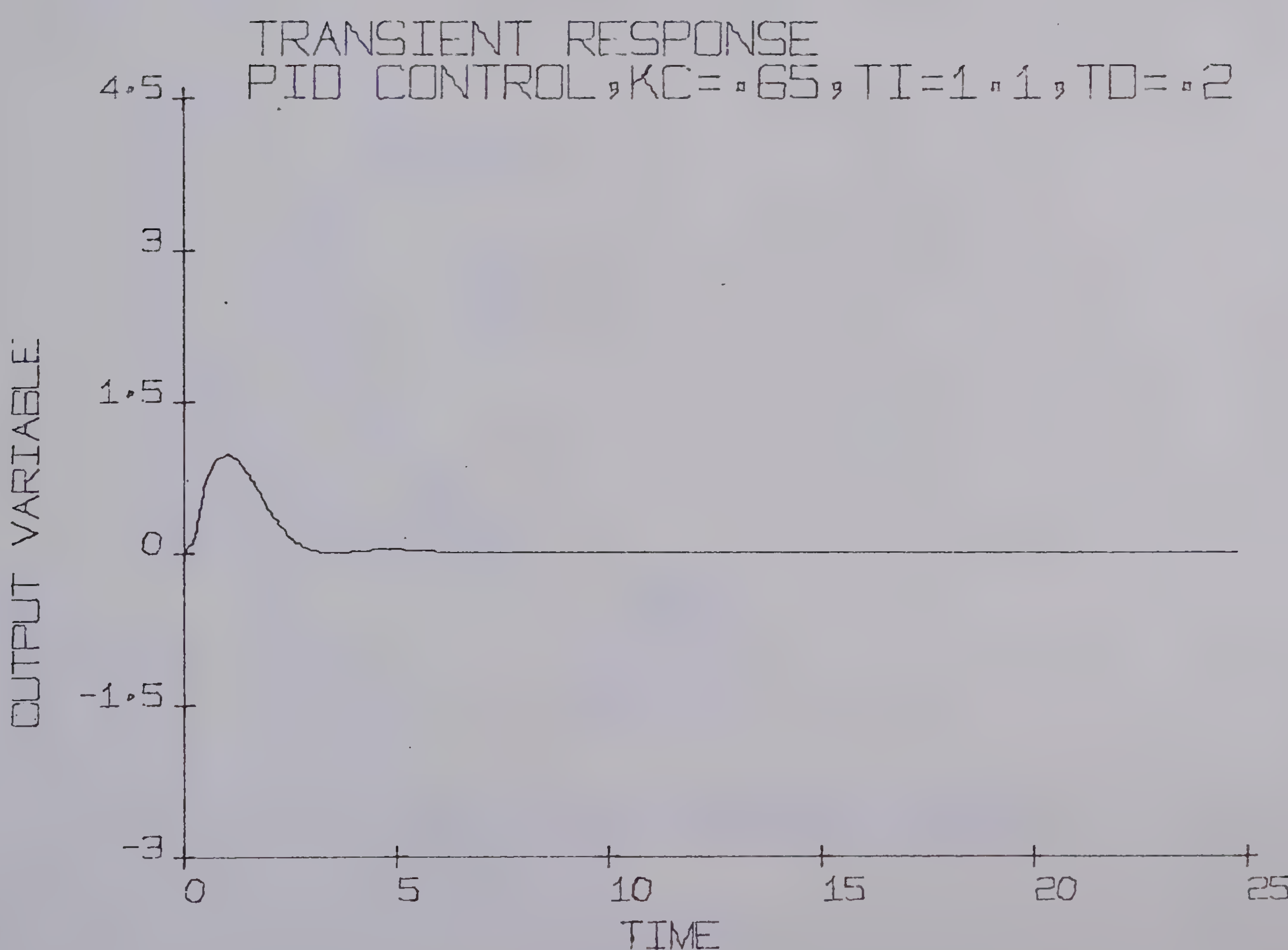
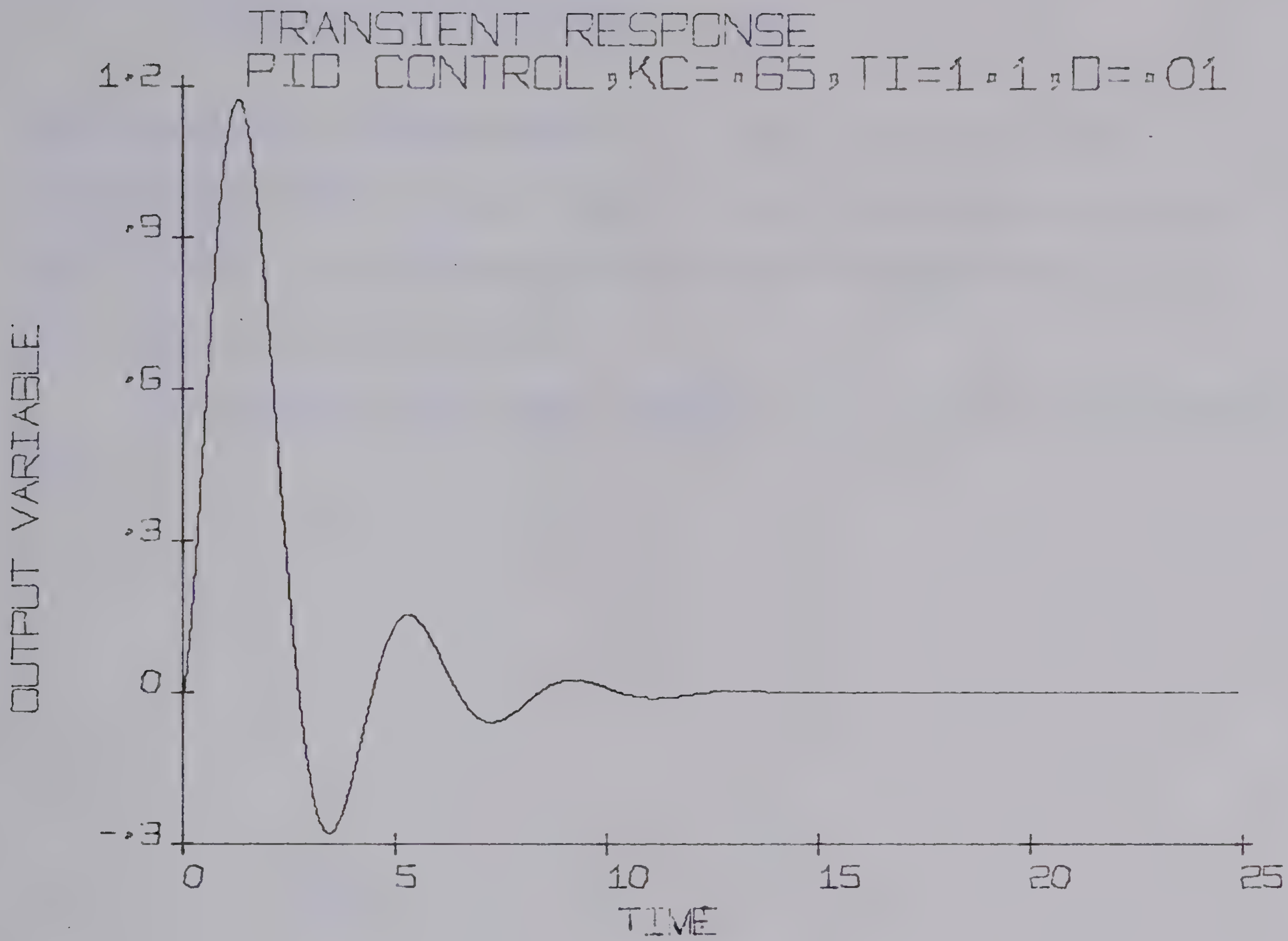


Figure 7-14. Variable Derivative Action Control.



PERFORMANCE CRITERIA FOR ABOVE RESPONSE

STEADY STATE	=	0.0000
PERCENT OVERSHOOT	=	0.00
PEAK TIME	=	1.3026
RISE TIME 0 TO 100	=	0.0001
RISE TIME 10 TO 90	=	0.0001
DELAY TIME	=	0.0000
DECAY RATIO	=	0.1323
SETTLING TIME	=	11.8236

ERROR INTEGRALS

IAE	=	2.46658
ISE	=	1.74813
ITAE	=	5.73353

Figure 7-15. Proportional-Integral-Derivative Control.

the derivative constant equal to 0.01 the plots of the Bode diagram, the Nyquist diagram, the Log Modulus diagram and the root locus diagram are shown in Figure 7-16, 7-17, 7-18, and 7-19 respectively.

The program was then terminated by entering a CONTROL DIGIT of 9.

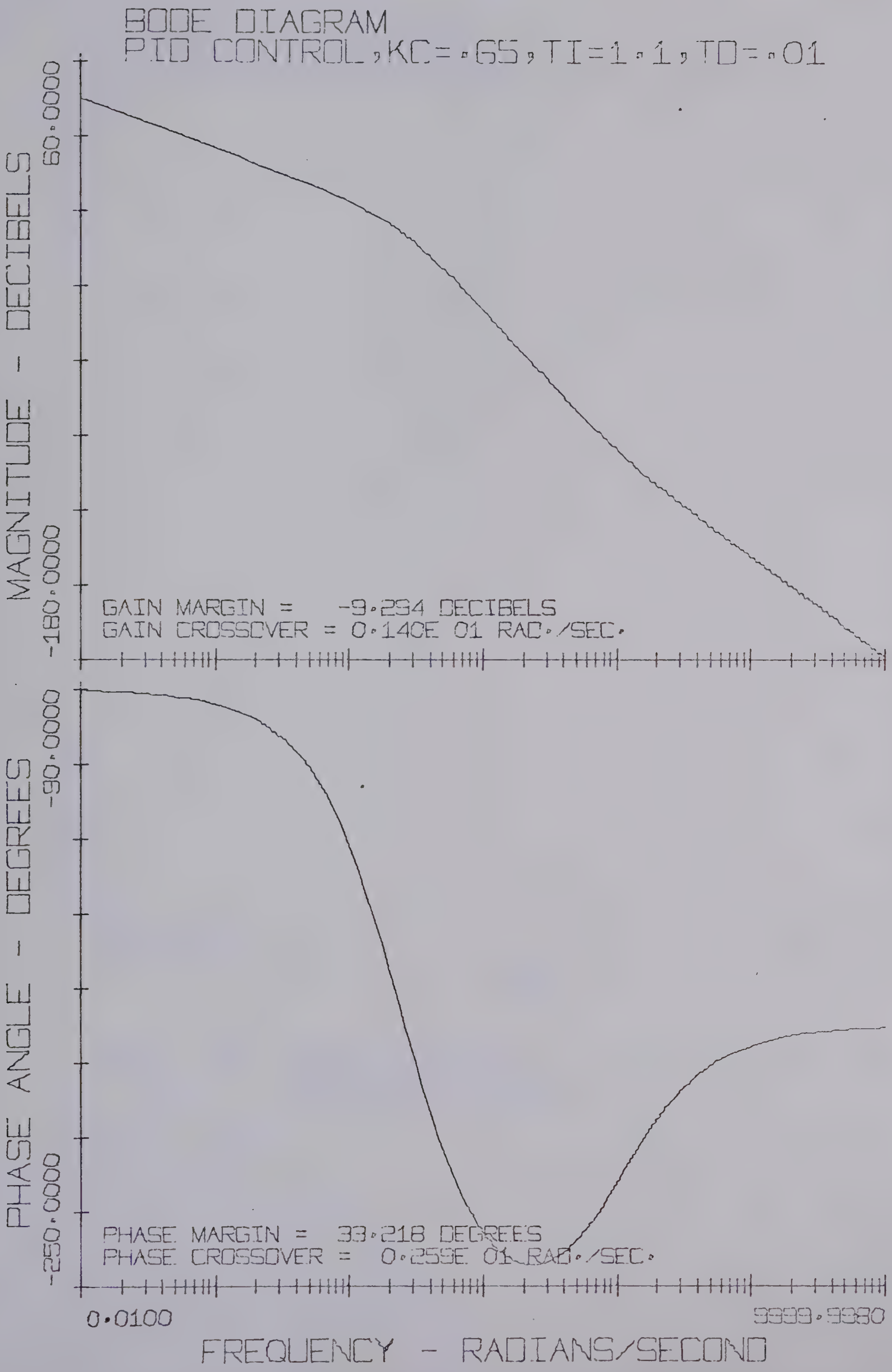


Figure 7-16. Bode Diagram For Proportional-Integral-Derivative Control.

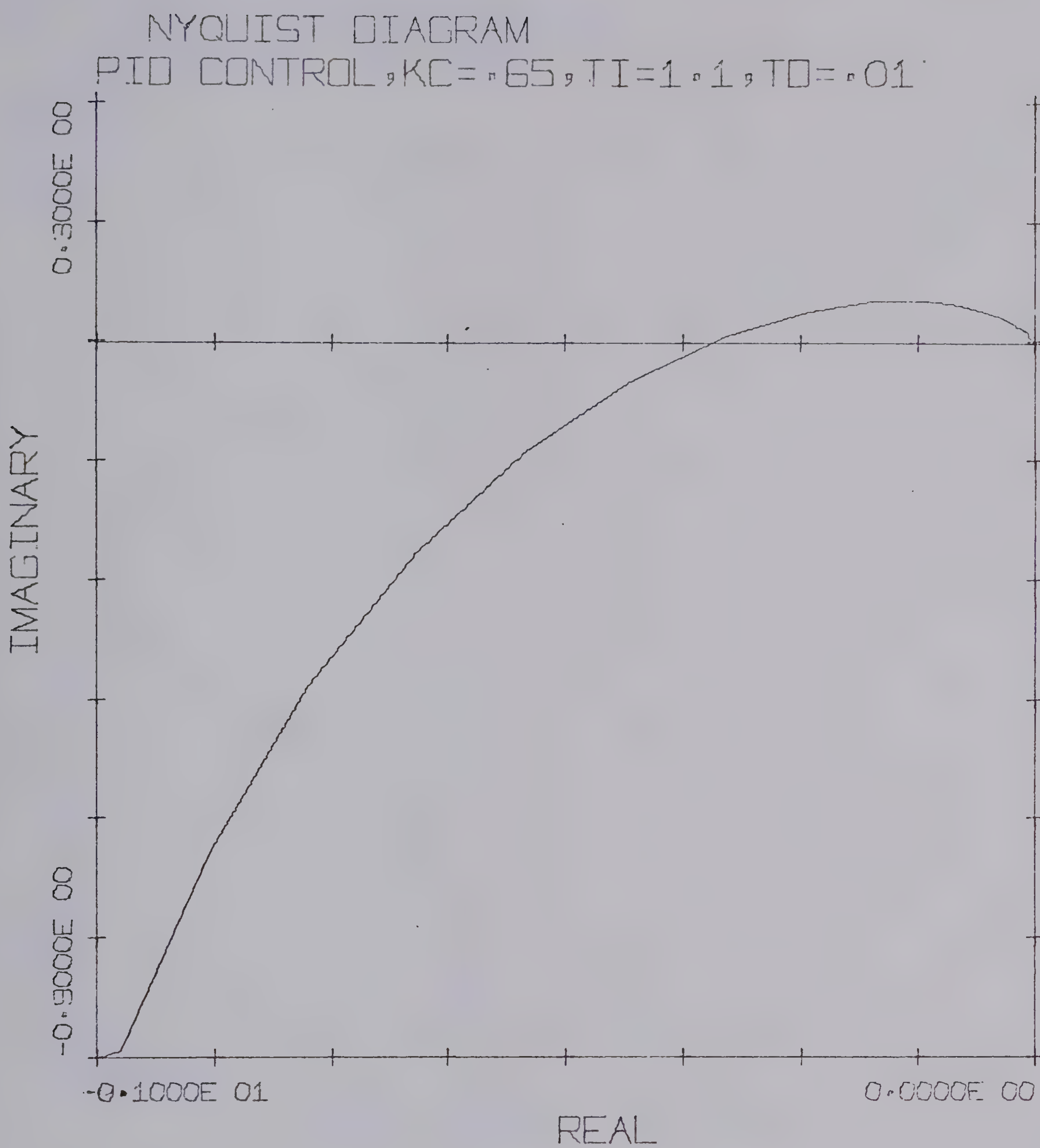
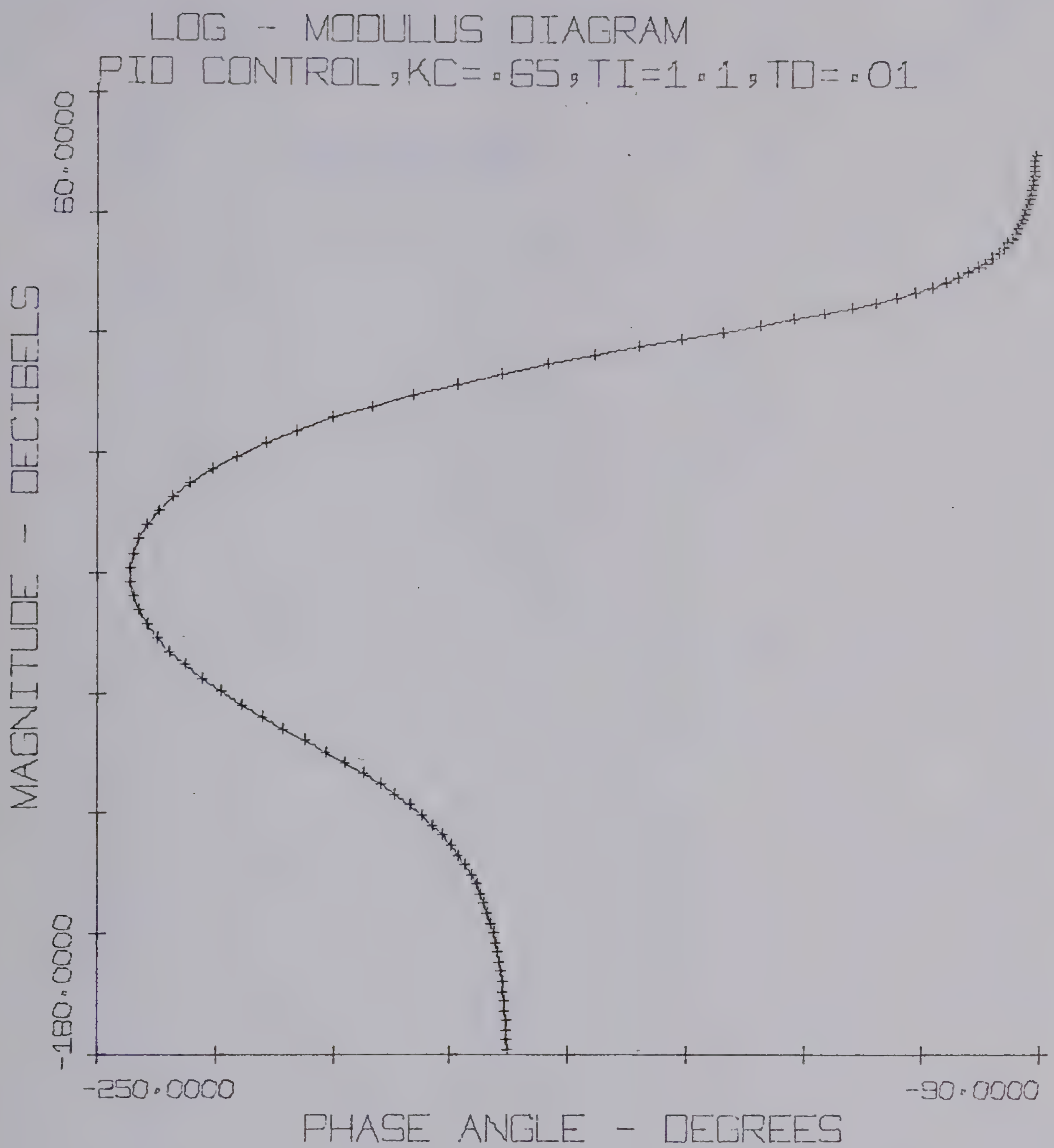


Figure 7-17. Nyquist Diagram For Proportional-Integral-Derivative Control.



PERFORMANCE CRITERIA FOR ABOVE DIAGRAM

PHASE CROSSOVER	=	2.5955	RAD./SEC.
GAIN MARGIN	=	-9.2947	DECIBELS
GAIN CROSSOVER	=	1.4030	RAD./SEC.
PHASE MARGIN	=	33.2188	DEGREES
PEAK RESONANCE	=	1.8177	DECIBELS
RESONANT FREQUENCY	=	1.4955	RAD./SEC.

Figure 7-18. Log-Modulus Plot For Proportyional-Integral-Derivative Control.

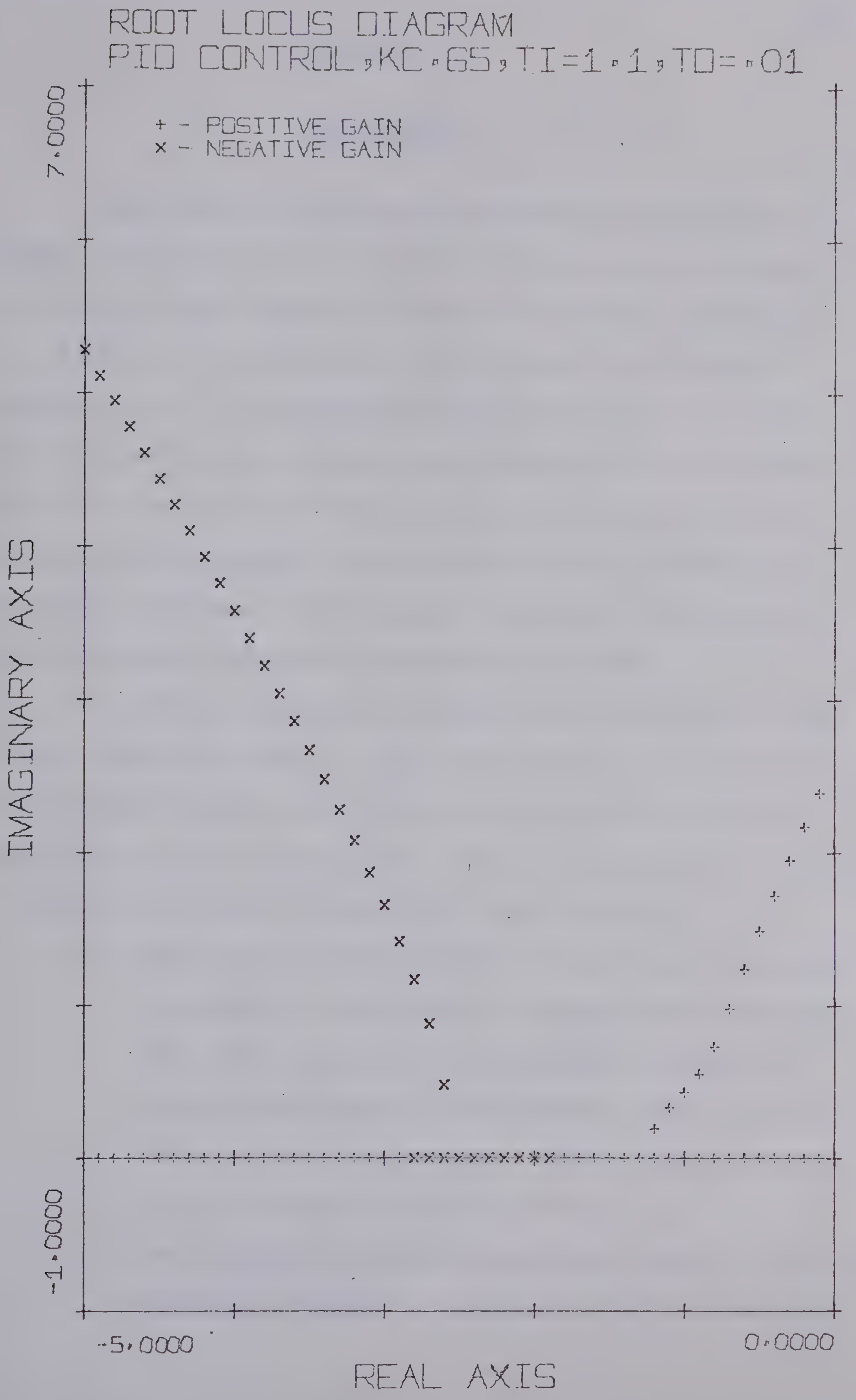


Figure 7-19. Root Locus Diagram For Proportional-Integral-Derivative Control.

CHAPTER VIII

CONCLUSIONS

The CONTROL SYSTEM DESIGN AND ANALYSIS PROGRAM (CSDAP) presented allows the user to design and analyze continuous linear feedback control systems that can be represented in the familiar block diagram and transfer function form. This system provides the control engineer with an easily used program that operates in a time sharing mode on a medium sized computer. Most conventional control systems design and analysis techniques using the transient response, the frequency response, and the root locus diagrams have been implemented in CSDAP.

The examples given in Chapter VII, Appendices A and B, and throughout some of the other chapters confirm that the original system design specifications for the CSDAP system have been accomplished. Some of the specific features of CSDAP that should be noted are the following:

(i) The program provides for completely documented problems in the form of a typewriter listing of the user program correspondence, calculated values available in the form of tables on the line printer, and graphs of the response curves on the Calcomp digital plotter.

(ii) The program itself provides the user with a considerable amount of assistance as data switch

options in all sections of the program. The user can gradually reduce this assistance as he acquires more experience with the use of the CSDAP system.

(iii) Individual block transfer functions can be modified without re-entering the whole control system configuration.

(iv) The user is provided with the ability to list all or individual transfer functions to the typewriter or line printer as he desires.

(v) The fact that the program has been developed in sections allows the user to alter problem specifications and parameters before rerunning the system analysis. For example, this means that the user can respecify the desired axis co-ordinates for a Nyquist diagram without recalculating the frequency response data values.

(vi) The effect on the control system behavior of a number of forcing functions when applied to the input variable of the system can be studied. Forcing functions that are presently implemented include the step, pulse, sine wave, ramp and impulse. Possible input variables to the system include the setpoint and the two load variables.

(vii) Transient response values can be calculated from the open loop transfer function or from the closed

loop transfer function when the output variable is taken as a signal either before or after the feedback element of the outer control loop.

(viii) The program allows a time delay to be included in either the forward or the feedback path of the control loop by using the Pade and Taylor series expansion approximation methods. The transient response data can be displayed in the form of a table of values on the line printer or as a plot on either the digital plotter or the storage oscilloscope.

(ix) A number of performance criteria such as the steady-state value, percent overshoot, peak time, rise time, delay time, decay ratio, settling time, and the error integrals associated with the transient response data can be calculated and displayed on the digital plotter, the storage oscilloscope or the typewriter.

(x) The program generates the frequency response data and displays it in the form of a Bode, Nyquist, Log Modulus or Root Locus diagram or lists it in the form of a table of values on the line printer.

(xi) The frequency response criteria can be calculated from the data stored on the disk. The gain and phase margins and crossover values are displayed on the Bode diagram. In addition to these values, the peak resonance and the resonant frequency are listed on the Log Modulus plot.

(xii) Options are present for replotting previously calculated transient response, frequency response, and root locus data that was stored in the disk files. The structure of these data files is described in the Systems Manual and should allow other programs to calculate and store the data in these files for use by the CSDAP system.

CHAPTER IX

RECOMMENDATIONS

Experience gained while using the program to generate the examples shown in this thesis has suggested a number of changes and additions that might be desirable for better program performance.

The first modification is concerned with the entry of transfer function data. The program would be changed so that a denominator factor would not need to be entered by the user unless present in the transfer function being entered. If not entered it would always contain unity. This modification would be particularly useful when studying the effect of proportional gain on the performance of a control system.

Another modification might be to include a step of unity as a standard forcing function. The system presently allows a unit step to the setpoint to be called for when entering the input variable of interest.

The third possible modification would be to change the method of calculating the steady state value. The Final Value Theorem would be used at the time the overall system transfer function was determined instead of the present method of determining an exponential envelope that encloses the transient response curve. The method that has been implemented works quite satisfactorily as long as the response curve is quite oscillatory.

An addition that might be helpful would be the inclusion of the values of the poles and zeros of the open loop transfer function on the root locus diagram. These values are not determined by the present program but should not be difficult to include. Some of the other rules that apply when constructing a root locus diagram such as the asymptotic behavior, the angle of departure or approach to the real axis, and the breakaway point could be implemented.

The program could be modified to include the z-transform capability by modifying the transfer function entry section of the CSDAP system. Program logic available in Lofkrantz's work (24) could then be used. A possible method of implementation would be to require the user to enter a z in front of the two transfer function name characters when it is desired to enter the transfer function as a polynomial in z notation.

Another special type of transfer function entry similar to that for a time delay could be used for entry of zero, first, and second order samplers once the system was modified to handle z-transforms.

BIBLIOGRAPHY

1. Agostinis, W., M.Sc. Thesis, Department of Chemical and Petroleum Engineering, University of Alberta, Edmonton, Alberta, 1969.

2. Buckley, P.S., "Techniques of Process Control", John Wiley and Sons, Inc., New York, N.Y., 1964.

3. Chang, C.S., "An Analytical Method for Obtaining the Root Locus with Positive and Negative Gain", IEEE Trans. on Automatic Control, No. 1, 10, 92 (1965).

4. Chen, C.F., and Haas, I.J., "Elements of Control Systems Analysis: Classical and Modern Approaches", Prentice-Hall, Inc., Englewood Cliffs, N.J.

5. Chen, C.F., and Shieh, L.S., "Generalization and Computerization of the Heaviside Expansion for Performing the Inverse Laplace Transform of High Order Systems", IEEE Region III Convention Record, 285-294, Mar., (1967).

6. Chen, C.F., and Yates, R.E., "A New Matrix Formula for the Inverse Laplace Transformation", ASME Trans., Journal of Basic Engineering, 269-272, June, (1967).

7. Churchill, R.V., "Operational Mathematics" 2nd ed., McGraw-Hill Book Company, New York, N.Y., 1958.

8. Coffey, T.C., "The Application of Modern Computing Technology to Control Systems Analysis and Design Problems", Aerospace Corp., El Segundo, Calif., AD 660 133, June (1967).

9. Conte, S.D., "Elementary Numerical Analysis - An Algorithmic Approach", McGraw-Hill Book Company, New York, N.Y., 1965.

10. Corrington, M.S., "Simplified Calculation of Transient Response", Proc. IEEE, 53, 287-292, March, (1965).

11. Coughanowr, D.R., and Koppel, L.B., "Process Systems Analysis and Control", McGraw-Hill Book Company, New York, N.Y., 1965.

12. Cummins, J.D., "A General Digital Computer Procedure for Synthesizing Linear Automatic Control Systems", Dynamics Group, Control and Instrumentation Division, A.E.E., Winfrith, Oct., 1961.

13. D'Azzo, J.J., and Haupis, C.H., "Feedback Control System Analysis and Synthesis", McGraw-Hill Book Company, New York, N.Y., 1966.

14. Evans, W.R., "Control-System Dynamics", McGraw-Hill Book Company, Inc., New York, N.Y., 1954.

15. I.B.M., "Control Systems Analysis", Program Reference Manual 7090/7094 (7090-MA-01X).

16. I.B.M., "1800 Time-Sharing Executive System Operating Procedures", Program Number 1800-OS-001-Version 3, File No. 1800-36, Form C26-3754-3.

17. I.B.M., "1130 Scientific Subroutine Package", 1130-CM-02X, Programmer's Manual (H20-025203).

18. I.B.M., "1130/1800 Basic FORTRAN IV Language", File No. 1130/1800-25, Form C26-3715-2.

19. I.B.M., "1800 Assemble Language", File No. 1800-21, Form C26-5882-4.

20. I.B.M., "1130/1800 Plotter Subroutines", File No. 1130/1800-30, Form C26-3755-1.

21. Krishnan, V., "Semi-Analytic Approach to Root Locus", IEEE Trans. A.C., Vol. AC-11, No. 1, January (1966).

22. Kuo, B.C., "Automatic Control Systems", 2nd Ed., Prentice-Hall Inc., Englewood Cliffs, N.J., 1967.

23. Liou, M.L., "A Novel Method of Evaluating Transient Response", Proc. IEEE, 54, 20-23, January, (1966).

24. Lofkrantz, J.E.E., M.Sc. Thesis, Department of Chemical and Petroleum Engineering, University of Alberta, Edmonton, Alberta, 1967.

25. Lopez, A.M., Miller, J.A., Smith, C.L., and Murrill, P.W., "Tuning Controllers with Error Integral Criteria", Instrumentation Technology, November, (1967).

26. Murphy, G.J., "Control Engineering", D. Van Nostrand Company, Inc., Princeton, N.J., 1959.

27. Murrill, P.W. and Smith, C.L., "Controllers - Set Them Right", Hydrocarbon Processing, No. 2, 45. February (1966).

28. Raven, F.H., "Automatic Control Engineering", 2nd Ed., McGraw-Hill Book Company, New York, N.Y., 1968.

29. Saucedo, R., and Schiring, E.E., "Introduction to Continuous and Digital Control Systems", MacMillan New York, 1968.

30. Smith, C.L., and Murrill, P.W., "A More Precise Method for Tuning Controllers", ISA Journal, May, (1966).

31. Truxal, J.G., "Automatic Feedback Control System Synthesis", McGraw-Hill Book Company, Inc.,New York, N.Y., 1954.

32. Wojcik, C.K., "Analytical Representation of Root Locus", J. Basic Eng., 37-43, March (1964).

33. Wylie, C.R. Jr., "Advanced Engineering Mathematics", 2nd Ed., McGraw-Hill Book Company, Inc., New York, 1960.

Appendix A

USER'S MANUAL

Section	Title	Page

List of Figures

List of Tables

Appendix A

USER'S MANUAL

A.1. Introduction

This section is intended to be the user's manual for the Control Systems Design and Analysis Program (CSDAP). This manual is divided into five sections that describe the five major operating sections of the CSDAP system.

The first is concerned with bringing the program into operation. After initialization it is necessary to enter configuration data for the control system to be studied and this is the concern of the second section. The last three sections outline the calculational procedure for determining the transient response, frequency response, and root locus for the control system.

Numerous tables and figures are provided throughout this description to clarify the specific operating procedures being discussed in each section.

A.2. Computer Operating Instructions

The following description concerning the use of the CSDAP system will proceed assuming that all necessary programs and data files have been successfully loaded onto the system. If this is not the case, the Systems Manual should be consulted for procedures to follow in loading the CSDAP system.

A.2.1. Run Initiation

This system as is described here is implemented on the IBM 1800 Data Acquisition and Control Computer. It is run as a Nonprocess Program under the IBM 1800 Timesharing Executive System (TSX). To initiate the program it is only necessary to load the following cards into the card reader and depress the Console Interrupt button (same procedure for any other Nonprocess Program).

```
    4        10      15     20         40
//  JOB       X       X      X          USER NAME

    4       8        16
//  XEQ     CSDAP    FX

//  JOB       X       X      X

    4
//  END     OF       ALL    JOBS

    BLANK   CARD
```

A.2.2. Operating Instructions

The next three subsections outline the operating procedures for the CSDAP system. They are general and apply to all phases of the CSDAP system and are given in this section rather than being repeated in the other sections.

A.2.2.1. Control Digit Entry

The control systems design and analysis program (CSDAP) is divided into nine logically self-contained sections. The entry into each of these sections is governed by the master control or executive program. This executive requests the user to enter the CONTROL DIGIT for the next section to be executed. A return is made to this executive after the completion of each section. Table A-1 lists the functions performed when each CONTROL DIGIT is entered.

A listing of this table can be obtained at the user's typewriter by having data switch 1 in the "ON" position when the program is initiated into execution. At any time after that the listing of the CONTROL DIGIT functions can be obtained by having this same data switch in the "ON" position and entering a CONTROL DIGIT of 0. The listing may be terminated by resetting the data switch to the "OFF" position.

A.2.2.2. Data Switch Functions

The user may need comprehensive input help to proceed to use this program the first few times. Later on this input help, depending on the sophistication of the user, may or may not be appreciated. If a user is very adept, excessive verbosity on the part of the program will be considered a severe annoyance and will make the

operation slow. If the user is a novice or highly subject to making errors, the help is appropriate to insure proper input.

The option to obtain assistance or not has been provided in CSDAP by means of data switches. The set of switches used depends on the setting of Sense Switch 2 on the IBM 1800 operator's console. If this switch is set in the "OFF" position, the console data switches will be used but if it is set in the "ON" position it causes the set of digital input switches to be used.

The data switches of the set chosen above when set in the "ON" position cause the program to list all possible alternatives or options available for the function parameter before it is requested to be entered. Table A-2 lists the functions of each data switch.

In general it can be said that anytime the user wants more than a single line of information when the input request is made, the pertinent data switch should be set to the "ON" position.

All data switch settings are updated by the executive program immediately after a legal CONTROL DIGIT is entered. This means that there will be time for the user to correctly set these switches before proceeding to the next program section.

TABLE A-1 CONTROL DIGIT INFORMATION

CONTROL DIGIT FUNCTIONS

0 = LIST CONTROL DIGIT AND DATA SWITCH OPTIONS
1 = INPUT TRANSFER FUNCTION DATA
2 = CALCULATE INVERSE LAPLACE TRANSFORM
3 = PLOT THE TIME RESPONSE DATA
4 = CALCULATE THE PERFORMANCE CRITERIA
5 = FREQUENCY RESPONSE CALCULATIONS
6 = ROOT LOCUS PLOT
7 = LIST TRANSFER FUNCTION AREA
8 = INITIALIZE TRANSFER FUNCTION AREA
9 = CALL EXIT

TABLE A-2 DATA SWITCH INFORMATION

DATA SWITCH FUNCTIONS

1 = LIST CONTROL DIGITS AND SWITCH OPTIONS
2 = LIST TRANSFER FUNCTION DATA AFTER INPUT
3 = INPUT-OUTPUT VARIABLE AND FORCING FUNCTION SELECTION
4 = LIST PARTIAL FRACTION DATA
5 = REPLOT, AND PLOT OPTIONS FOR TIME RESPONSE
6 = LIST TIME RESPONSE DATA ON THE PRINTER
7 = PERFORMANCE DATA OUTPUT DEVICE OPTIONS
8 = FREQUENCY RESPONSE DATA OPTIONS
9 = BODE PLOT OPTIONS
10 = LIST FREQUENCY RESPONSE DATA ON THE PRINTER
11 = NYQUIST PLOT OPTIONS
12 = LOG-MODULUS PLOT OPTIONS
13 = ROOT LOCUS PLOT OPTIONS
14 = LIST ROOT LOCUS DATA ON THE PRINTER

An example of the additional information that is typed out to the user is illustrated in Figure A-1. The first section shows the correspondence between the user and the program if data switch 3 was in the "ON" position. The same data entries are made in the second part of the figure but now data switch 3 is in the "OFF" position. This gives an example of the varying amount of support that is available to the user.

A.2.2.3. User Data Input

The program uses two subroutines written in assembly language for ease in entering the input data. The first of these allows entry of a single numeric digit from the typewriter keyboard. This was necessitated because many of the parameter options could be coded by using some contiguous string of numeric digits, 0-9. The end-of-field character does not have to be typed in addition to the desired digit therefore improving the speed of operation. A legality check for a numeric character within the desired range is made internal to the subroutine and input is requested again if in error. This is illustrated in the following statements taken from a typical run of CSDAP. The user's input to the program request is underlined.

```
     ENTER CONTROL DIGIT
2
      ENTER TRANSIENT RESPONSE OF INTEREST
      0 = OPEN LOOP
      1 = CLOSED LOOP BEFORE FEEDBACK
      2 = CLOSED LOOP AFTER FEEDBACK
1
      ENTER INPUT VARIABLE
      1 = SETPOINT
      2 = LOAD ONE
      3 = LOAD TWO
      4 = UNIT STEP TO SETPOINT
1
      ENTER INPUT VARIABLE FORCING FUNCTION
      1 = STEP
      2 = PULSE
      3 = SINE
      4 = RAMP
      5 = IMPULSE
2
      ENTER PULSE HEIGHT AND TIME DURATION
1.0 2.0

     ENTER CONTROL DIGIT
3
```

FIGURE A-1A EXTENSIVE USER ASSISTANCE FROM CSDAP SYSTEM

```
     ENTER CONTROL DIGIT
2
      ENTER TRANSIENT RESPONSE OF INTEREST
1
      ENTER INPUT VARIABLE
1
      ENTER INPUT VARIABLE FORCING FUNCTION
2
      ENTER PULSE HEIGHT AND TIME DURATION
1.0 2.0

     ENTER CONTROL DIGIT
3
```

FIGURE A-1B LIMITED USER ASSISTANCE FROM CSDAP SYSTEM

```
          ENTER TRANSIENT RESPONSE OF INTEREST
          0 = OPEN LOOP
          1 = CLOSED LOOP BEFORE FEEDBACK
          2 = CLOSED LOOP AFTER FEEDBACK
    I
          INPUT IS IN ERROR, TRY AGAIN
    3
          INPUT IS IN ERROR, TRY AGAIN
    2
          ENTER INPUT VARIABLE
            .
            .
            .
```

The first error was caused because a non-numeric character was entered while the second error resulted because the numeric character entered was not in the range 0 to 2.

The other subroutine allows the CSDAP user a less restrictive method of data input than would be possible if the program were written in FORTRAN. Integer and real variables in addition to alphanumeric data are read in as single words or into vectors. The variables need only be separated by a space or comma instead of being right justified in their respective fields as required by a FORTRAN request. Leading and/or trailing blanks are ignored with the decimal point of floating point numbers being necessary

only when a fractional part is present.

This subroutine also performs a check on the input data and results in notification of the calling program for suitable action, usually to request the data again. The following few lines show the use of this subroutine with the user's reply underlined.

```
                ENTER INITIAL AND TERMINAL FREQUENCIES
CSDAP
                INPUT IN ERROR, TRY AGAIN
-8
                INPUT IN ERROR, TRY AGAIN
-8  2.5
```

The first error resulted because non-numeric data was entered for the data required. The last error was caused because only one of the two frequency values was entered correctly.

A.2.3. Run Termination

To terminate operation of the CSDAP system, the user simply enters a CONTROL DIGIT of 9 which causes the program to give control back to the Nonprocess Monitor of the IBM 1800 TSX system.

A.3. Transfer Function Specification

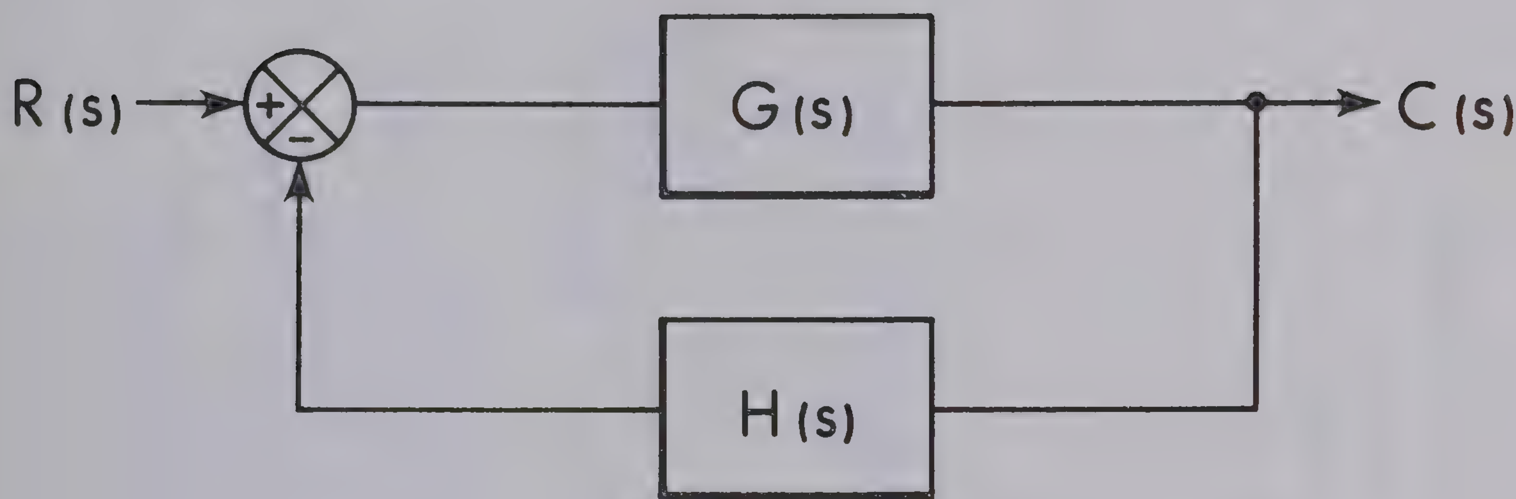
A.3.1. Control System Representation

The analysis of linear feedback control systems is facilitated by the use of transfer functions and block diagrams. The block diagram is a functional representation of an entire control system with the component transfer functions shown within the blocks and the flow of signals shown by lines connecting the blocks. Each block is an independent unit in that the connection between the blocks does not affect the transfer function inside them.

A simple single loop control system is shown in Figure A-2. It serves to show some of the special transfer functions that can be calculated from the two blocks. Seldom is it possible to represent a control system with such a simplified block diagram. The control engineer is usually interested in analyzing the system performance for a number of input and output variables. Consequently, to handle the majority of systems that would be of interest, it was decided that the general system representation as shown in the block diagram of Figure A-3 would be employed.

A.3.2. Block Diagram Transfer Functions

To facilitate easy block diagram manipulation, the transfer function data for each block of Figure A-3 is



NOMENCLATURE

R(s)	- Setpoint or reference input
C(s)	- Output or controlled variable
G(s)	- Forward path transfer function
H(s)	- Feedback path transfer function
G(s) H(s)	- Open loop transfer function
$\dfrac{G(s)}{1 + G(s)\,H(s)}$	- Closed loop transfer function
$1 + G(s)\,H(s) = 0$	- Characteristic equation

FIGURE A-2 Simple Control Loop System

NOMENCLATURE

R(s)	-	Setpoint or reference input
C(s)	-	Output or controlled variable
B(s)	-	Feedback signal
U1(s), U2(s)	-	Load variables or disturbances
C1(s), C2(s)	-	Controller transfer functions
P1(s), P2(s), Ps(s)	-	Process transfer functions
H1(s), H2(s)	-	Feedback element transfer functions
L1(s) L2(s)	-	Load transfer functions

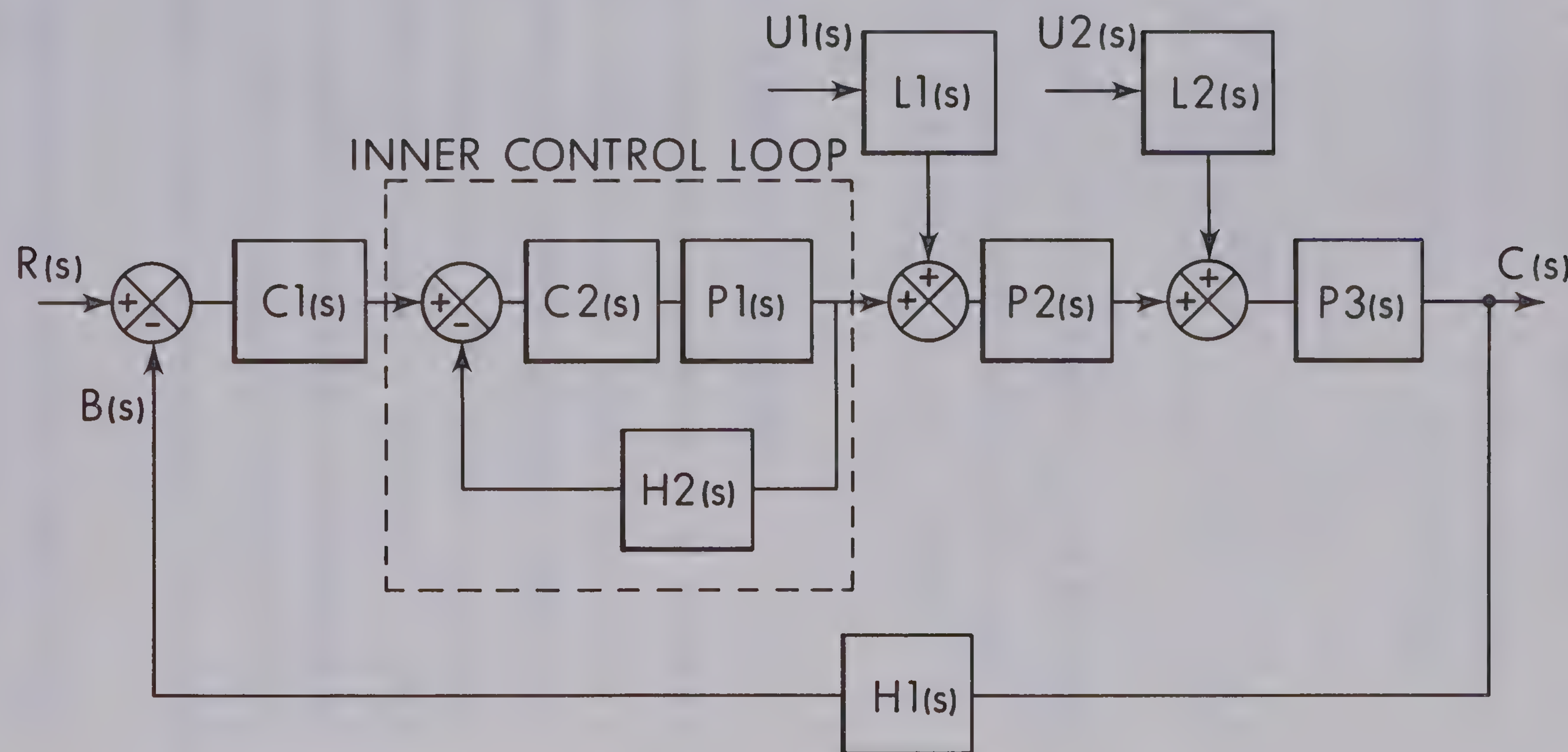


FIGURE A-3 General Control System Representation

entered as a ratio of polynomial terms in the complex variable s except for a time delay. The method of entering a time delay is explained in a later section. A transfer function can be described by the following general expression.

$$ID(s) = \frac{(a_1 s + a_2)(a_3 s^2 + a_4 s + a_5)(\ldots) \ldots}{(b_1 s + b_2)(b_3 s^2 + b_4 s + b_5)(\ldots) \ldots} \qquad \text{(A-1)}$$

The transfer function is specified in the program by the two alphanumeric characters ID which can be any of those shown in Figure A-3 such as P2, C1, or H2 to mention a few.

A.3.3. Transfer Function Data Entry

The coefficients for each polynomial of a transfer function, of the general form shown in Equation (A-1), are entered within brackets and appear ordered from the smallest to the largest power of s. The first value is always the constant term (coefficient of the zero power of s). The distinction between numerator and denominator polynomials is made by placing a slash or division sign after the closing bracket of the last numerator polynomial specification.

Input data for specification of the transfer function given as Equation (A-1) is entered by the user as:

$$ID(a_2,a_1)(a_5,a_4,a_3)\ \ldots/(b_2,b_1)(b_5,b_4,b_3)\ \ldots \qquad \text{(A-2)}$$

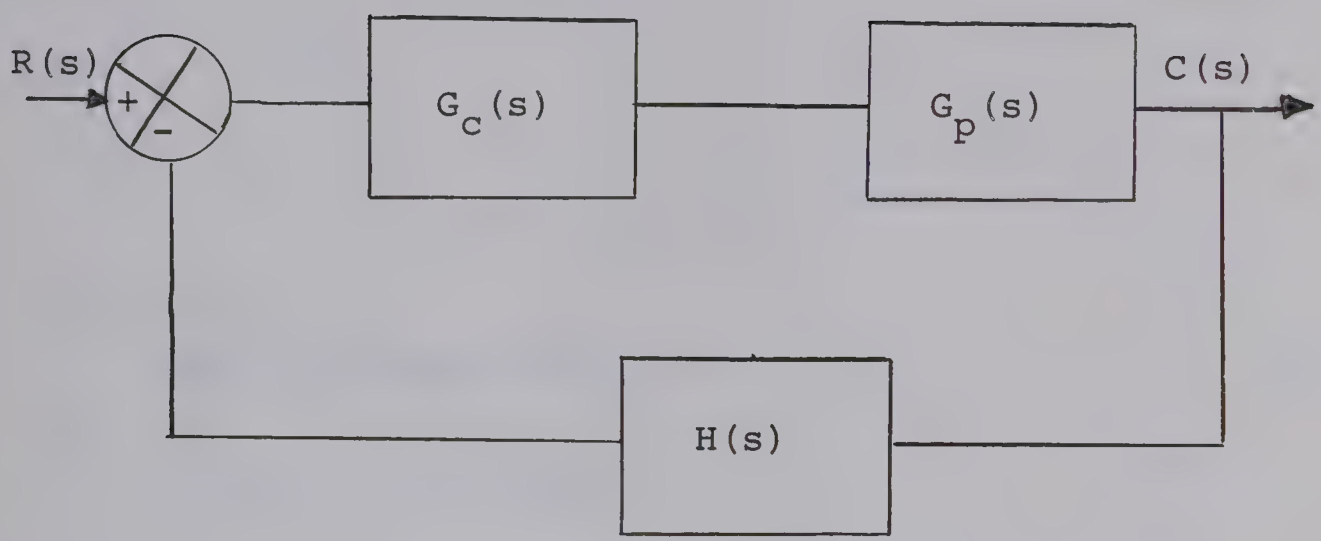
To configurate a typical control system the transfer functions of a number of the blocks must be entered. The order of entering the block diagram transfer functions is irrelevant. It is important to realize that the program is written in such a manner that whenever a new transfer function is entered correctly it automatically replaces the earlier version of the same transfer function.

The data is checked for a legal transfer function (block) name and the correct coefficient arrangement immediately after input. If the data passes the check it is transferred to permanent storage. If an error is found a suitable error message is typed to the user depending on the type of error.

If there is a name error the input section of the program will return to the executive to request another control digit. This provides the means for exiting from this section of the program. When an error in entering the input data occurs it is assumed that the user is serious about entering this transfer function data and the program requests that the data be re-entered.

Figure A-4 shows the information the user had to supply to CSDAP for configurating the control system

given below.



where

$$G_p(s) = \frac{2.5\ (s + 1.1)}{s(1 + 0.25s)\ (s + 1)}$$

$$G_c(s) = K_c(1 + 1/\tau_I s)$$

$$= K_c \frac{\tau_I s + 1}{\tau_I s} \text{ or } K_c \frac{s + 1/\tau_I}{s}$$

$$= 1.25 \frac{(0.33s + 1)}{0.33s}$$

$$H(s) = \frac{1}{0.5s + 1}$$

For the entry of these transfer functions the user can let C1(s) = $G_c(s)$, P2(s) = $G_p(s)$, and H1(s) = H(s).

```
     ENTER CONTROL DIGIT
8

     ENTER CONTROL DIGIT
1

      ENTER TRANSFER FUNCTION DATA
P2(2.5)(1.1,1)/(0,1)(1,0.25)(1,1)

      ENTER TRANSFER FUNCTION DATA
C1(1.25)(1,0.33)/N/,/.OOE
      ERROR IN INPUT DATA - TRY AGAIN
C1(1.25)(1,0.33)/(0,0.33)

      ENTER TRANSFER FUNCTION DATA
H1(1)/(1,0.5)

      ENTER TRANSFER FUNCTION DATA

      ILLEGAL TRANSFER FUNCTION NAME

     ENTER CONTROL DIGIT
2
```

FIGURE A-4 CONFIGURATION OF A CONTROL SYSTEM

A.3.4. Transportation Delay

The pure time delay or transportation lag, $e^{-\tau s}$, is frequently found in control systems. It cannot be expressed exactly as a ratio of polynomials as was the case for the other transfer functions previously discussed. It is therefore necessary to use an approximation method.

The program allows two approximation methods. One method involves the replacement of $e^{-\tau s}$ by a rational function of s such that the first terms in the Taylor power series expansion of $e^{\tau s}$ appear in the denominator as shown in Equation (A-3).

$$e^{-\tau s} = \frac{1}{e^{\tau s}} = \frac{1}{1 + s + \tau^2 s^2/2! + \tau^3 s^3/3! + \ldots} \quad \text{(A-3)}$$

In this program up to a fourth-degree Taylor series expansion can be used in the denominator.

The other method is known as the Padé approximation method. In this case the power series expansion is approximated by an equal number of poles and zeros. ne approximati improves as the number of poles and zeros used increases. In this program a Padé approximation with up to four poles and four zeros or in other words up to a fourth degree polynomial in both the numerator and denominator can be requested. The general fourth degree Padé approximation

is given in the following equation.

$$e^{-\tau s} = \frac{1 - \tau s/2 + \tau^2 s^2/12 - \tau^3 s^3/120 + \tau^4 s^4/1680}{1 + \tau s/2 + \tau^2 s^2/12 + \tau^3 s^3/120 + \tau^4 s^4/1680} \qquad \text{(A-4)}$$

To enter a time delay approximation the user enters a transfer function name of TD when requested to enter transfer function data. The program then requests the user to enter further information describing the desired time delay approximation.

The time delay can be placed in the forward path, as block P3, or in the feedback path as block H1. Once the rational polynomial approximation is calculated it is multiplied by the existing contents of the respective block P3 or H1. This makes it possible to have another transfer function already present for this block and dictates that the time delay should be the last transfer function entry for these two particular blocks.

Two examples of the information that must be provided by the user for obtaining a time delay approximation in the control system are shown in Figures A-5A and A-5B. Figure A-5A shows how a third order Padé approximation could be placed in the forward path while Figure A-5B shows a fourth degree Taylor series approximation being configurated for the feedback transfer function H1(s).

```
     ENTER CONTROL DIGIT
1

      ENTER TRANSFER FUNCTION DATA
TD
      TIME DELAY - 0=FORWARD PATH,1 = FEEDBACK PATH
0
      APPROXIMATION TYPE - 0=PADE,1=POWER SERIES
0
      ENTER VALUE OF TIME DELAY
0.15
      ENTER ORDER OF PADE APPR. =1,2,3,4
3

      ENTER TRANSFER FUNCTION DATA
C1(2.5)...
```

FIGURE A-5A THIRD ORDER PADE IN FORWARD PATH

```
     ENTER CONTROL DIGIT
1

      ENTER TRANSFER FUNCTION DATA
TD
      TIME DELAY - 0=FORWARD PATH,1 = FEEDBACK PATH
1
      APPROXIMATION TYPE - 0=PADE,1=POWER SERIES
1
      ENTER VALUE OF TIME DELAY
0.15
      ENTER DEGREE OF POWER SERIES DESIRED 1,2,3,4
4

      ENTER TRANSFER FUNCTION DATA
C1(2.5)...
```

FIGURE A-5B FOURTH DEGREE TAYLOR SERIES IN FEEDBACK PATH

A.3.5. Data Input Checking

Configurating the control system to be studied must be done with the most care. This is because the operation of entering the transfer function data is most susceptible to errors. The input data is checked by the program for the correct transfer function name and the proper configuration of brackets. The coefficients themselves cannot be checked because only the user knows these values.

To facilitate additional checking of the parameters entered data switch 2, if set in the "ON" position, causes a transfer function to be immediately listed back to the user if it was successively entered. For the numerator the first line is an integer description that gives the number of polynomials that were found in the user supplied data plus the degree of each of these polynomials. The next line gives the coefficients for each polynomial successively. The coefficients are ordered from the lowest to the highest power of s in a similar manner to the transfer function input data. The denominator data is listed in the same format.

An example of this data listing is given in Figure A-6 which is the same transfer function input data as shown in Figure A-4 but because data switch 2 was set in the "ON" position, the additional listing of the transfer

```
     ENTER CONTROL DIGIT
8

     ENTER CONTROL DIGIT
1

      ENTER TRANSFER FUNCTION DATA
P2(2.5)(1.1,1)/(0,1)(1,0.25)(1,1)
          2     0     1
        2.50000   1.10000   1.00000
          3     1     1     1
        0.00000   1.00000   1.00000   0.25000   1.00000
        1,00000

      ENTER TRANSFER FUNCTION DATA
C1(1.25)(1,0.33)/N/,/.OOE
      ERROR IN INPUT DATA - TRY AGAIN
C1(1.25)(1,0.33)/(0,0.33)
          2     0     1
        1.25000   1.00000   0.33000
          1     1
        0.00000   0.33000

      ENTER TRANSFER FUNCTION DATA
H1(1)/(1,0.5)
          1     0
        1.00000
          1     1
        1.00000   0.50000

      ENTER TRANSFER FUNCTION DATA

      ILLEGAL TRANSFER FUNCTION NAME

     ENTER CONTROL DIGIT
2
```

FIGIRE A-6 DATA VERIFICATION AFTER INPUT

function polynomial coefficients is provided.

A.3.6. Listing of Transfer Functions

If at some time the user wants to know the current status of some or all of the transfer functions in the system, perhaps because of modifications to the original control configuration, this can be obtained.

This service is available by entering a CONTROL DIGIT of 7. The user specifies the output device, typewriter or line printer, where the transfer functions are to be listed and then enters the names of the transfer functions to be listed. If at any time the user specifies a transfer function name of 99, a complete listing of all the transfer functions will be made on the line printer. The transfer function names, and their relation to the black diagram of Figure A-3, are given in Table A-3.

The data that can be listed on either output device is given as a single polynomial formed from component polynomials of the transfer function. Again the coefficients appear in ascending powers of the s variable with the s^0 coefficient always appearing at the beginning.

Figure A-7 shows the entries to list some of the transfer function data and the result while Figure A-8 gives a complete listing of all the transfer function data.

TABLE A-3 TRANSFER FUNCTION NAMES

ABBREVIATION	DESCRIPTION	SOURCE
C1(S)	CONTROL BLOCK ONE	USER ENTERED
C2(S)	CONTROL BLOCK TWO	USER ENTERED
P1(S)	PROCESS BLOCK ONE	USER ENTERED
P2(S)	PROCESS BLOCK TWO	USER ENTERED
P3(S)	PROCESS BLOCK THREE	USER ENTERED
H1(S)	FEEDBACK BLOCK ONE	USER ENTERED
H2(S)	FEEDBACK BLOCK TWO	USER ENTERED
L1(S)	LOAD BLOCK ONE	USER ENTERED
L2(S)	LOAD BLOCK TWO	USER ENTERED
TD	TIME DELAY APPROXIMATION	USER ENTERED
OL(S)	OPEN LOOP	CALCULATED
CL(S)	CLOSED LOOP	CALCULATED
FP(S)	FORWARD PATH	CALCULATED
SP(S)	SETPOINT FORCING FUNCTION	CALCULATED
U1(S)	LOAD ONE FORWARD PATH	CALCULATED
LF(S)	LOAD ONE * FORCING FUNCTION	CALCULATED
U2(S)	LOAD TWO FORWARD PATH	CALCULATED
LS(S)	LOAD TWO * FORCING FUNCTION	CALCULATED
TM(S)	OVERALL TIME	CALCULATED

```
     ENTER CONTROL DIGIT
7
      ENTER LIST DEVICE 0=TYPEWRITER,1=PRINTER
0
      ENTER NAMES OF TRANSFER FUNCTIONS TO BE LISTED
      I.E. P1C1H1H2...   OR 99 TO LIST THEM ALL
CSDAP
      NO TRANSFER FUNCTION NAMES ENTERED
      ENTER 1 TO CONTINUE , 2 TO EXIT
1
      ENTER LIST DEVICE 0=TYPEWRITER,1=PRINTER
0
      ENTER NAMES OF TRANSFER FUNCTIONS TO BE LISTED
      I.E. P1C1H1H2...   OR 99 TO LIST THEM ALL
P2C1H1

           *** PROCESS TWO   ***

           NUMERATOR
           2.75000     2.50000
           DENOMINATOR
           0.00000     1.00000     1.25000     0.25000

           *** CONTROL ONE  ***

           NUMERATOR
           1.25000     0.41249
           DENOMINATOR
           0.00000     0.33000

           *** FEEDBACK ONE  ***

           NUMERATOR
           1.00000
           DENOMINATOR
           1.00000     0.50000

     ENTER CONTROL DIGIT
7
      ENTER LIST DEVICE 0=TYPEWRITER,1=PRINTER
1
      ENTER NAMES OF TRANSFER FUNCTIONS TO BE LISTED
      I.E. P1C1H1H2...   OR 99 TO LIST THEM ALL
99

      ENTER TITLE FOR TRANSFER FUNCTION LISTING
EXAMPLE LISTING FOR USER'S MANUAL
```

FIGURE A-7 TRANSFER FUNCTION LISTING OPTIONS

FIGURE A-8 LISTING OF TRANSFER FUNCTIONS
PAGE 1 OF 3

```
***  EXAMPLE LISTING FOR USER'S MANUAL          ***


*** CONTROL ONE    ***

NUMERATOR
1.25000      0.41249
DENOMINATOR
0.00000      0.33000

*** CONTROL TWO    ***

NUMERATOR
1.00000
DENOMINATOR
1.00000

*** PROCESS ONE    ***

NUMERATOR
1.00000
DENOMINATOR
1.00000

*** PROCESS TWO    ***

NUMERATOR
2.75000      2.50000
DENOMINATOR
0.00000      1.00000      1.25000      0.25000

*** PROCESS 3      ***

NUMERATOR
1.00000
DENOMINATOR
1.00000

*** FEEDBACK ONE   ***

NUMERATOR
1.00000
DENOMINATOR
1.00000      0.50000
```

FIGURE A-8 LISTING OF TRANSFER FUNCTIONS


```
***  EXAMPLE LISTING FOR USER'S MANUAL          ***


*** FEEDBACK TWO   ***

NUMERATOR
0.00000
DENOMINATOR
1.00000

*** LOAD ONE       ***

NUMERATOR
1.00000
DENOMINATOR
1.00000

*** LOAD TWO       ***

NUMERATOR
1.00000
DENOMINATOR
1.00000

*** OPEN LOOP      ***

NUMERATOR
3.43749      4.25937      1.03124
DENOMINATOR
0.00000      0.00000      0.33000      0.57749      0.28874
0.04124

*** CLOSED LOOP    ***

NUMERATOR
3.43749      5.97812      3.16093      0.51562
DENOMINATOR
3.43749      4.25937      1.36124      0.57749      0.28874
0.04124

*** FORWARD PATH   ***

NUMERATOR
3.43749      4.25937      1.03124
DENOMINATOR
0.00000      0.00000      0.33000      0.41249      0.08249
```

FIGURE A-8 LISTING OF TRANSFER FUNCTIONS
PAGE 3 OF 3

```
***  EXAMPLE LISTING FOR USER'S MANUAL          ***


*** SETPOINT FF   ***

NUMERATOR
3.43749     5.97812     3.16093     0.51562
DENOMINATOR
3.43749     4.25937     1.36124     0.57749     0.28874
0.04124

*** LOAD 1 FPATH  ***

NUMERATOR
1.00000
DENOMINATOR
1.00000

*** LOAD 1 * FF   ***

NUMERATOR
1.00000
DENOMINATOR
1.00000

*** LOAD 2 FPATH  ***

NUMERATOR
1.00000
DENOMINATOR
1.00000

*** LOAD 2 * FF   ***

NUMERATOR
1.00000
DENOMINATOR
1.00000

*** OVERALL TIME  ***

NUMERATOR
83.33335   144.92424    76.62878    12.49999
DENOMINATOR
0.00000    83.33335    103.25758    33.00000    14.00000
6.99999     1.00000
```

A.3.7. Initialization of the Transfer Function Area

All transfer functions are initialized when CSDAP is brought into operation or later by entering a CONTROL DIGIT of 8. This procedure sets the degrees of the numerator and denominator polynomials equal to zero and sets the constant coefficients associated with zero degree polynomials equal to unity except in the case of the feedback transfer function of the inner control loop, namely H2(s). Its numerator is set equal to zero to indicate zero feedback on this inner loop.

A.4. Transient Response Calculation

This section of the manual will describe three phases of the CSDAP that are utilized in performing the time response calculation. The first phase is used to calculate the overall control system transfer function and then to perform an inverse Laplace transformation of this transfer function. The second phase calculates and displays the time response data for the previously inverted overall transfer function. The third phase calculates and lists the performance criteria associated with the time response calculated in the second phase.

A.4.1. Transient Response of Interest

The transient response of the variable of interest is the first information that is requested since the program allows the use of different transfer functions. The transient response of the open loop transfer function, defined as the transfer function B(s)/R(s) with the feedback path on Figure A-3 broken at the summing junction, is one optional response that can be calculated. The input variable illustrated here as the setpoint R(s), can be the setpoint R(s) or either load variable.

The response of the closed loop transfer function considered as C(s)/R(s) or B(s)/R(s), as defined in Figure A-3, can also be obtained. The option of specifying either closed loop transfer function is included because although it is common to refer to C(s) as the output variable, this variable usually cannot be measured. The control signal is that from the measuring element transfer function H1(s), namely B(s).

Figure A-1 gives an example of the input to the CSDAP to specify the desired transfer function of interest.

A.4.2. Input and Output Variables

The open or closed loop transfer function is used to allow the design engineer to study the effects on the system behavior of changes in different input variables.

The procedure is to decide which of the possible inputs of the system to study and to call this the input variable. Another property will be called the dependent variable and may be considered the output. A relationship between the input and output variables is determined, and then the change in the input variable is introduced.

The setpoint, load one, or load two are available as input variables while the output variable can be either C(s) or B(s), as discussion in Section 4.2. The different types of disturbance that can be used for any of the input variables are shown in Table A-4, which in addition to the codes gives the additional information required for each type of disturbance. Examples of the information supplied to the CSDAP are given in Figures A-3 and A-9.

A.4.3. Inverse Laplace Transform Calculation

When the user has specified the desired input variable and the type of forcing function with its additional parameters, the program proceeds to calculate the overall transfer function. The inverse Laplace transformation is calculated from this transfer function by the method of partial fractions. The data vectors associated with the partial fraction coefficients can be optionally listed for the user by setting data switch 4 to the "ON" position. The data is in such a form that

TABLE A-4 FORCING FUNCTION CODES

TYPE	CODE NUMBER	FIRST DISTURBANCE PARAMETER	SECOND DISTURBANCE PARAMETER
STEP	1	MAGNITUDE	--
PULSE	2	PULSE HEIGHT	--
SINE	3	AMPLITUDE	PERIOD
RAMP	4	SLOPE	--
IMPULSE	5	--	--

```
     ENTER CONTROL DIGIT
8

     ENTER CONTROL DIGIT
1

      ENTER TRANSFER FUNCTION DATA
P2(1)/(1,1)(1,1.5)

      ENTER TRANSFER FUNCTION DATA
C1(200)/(3)(0,1)

      ENTER TRANSFER FUNCTION DATA
C2(0.75,1)/(1)

      ENTER TRANSFER FUNCTION DATA

      ILLEGAL TRANSFER FUNCTION NAME

     ENTER CONTROL DIGIT
3
      INVERSE TRANSFORMATION HAS NOT BEEN CALCULATED
      SINCE TRANSFER FUNCTION AREA CHANGED
      ENTER CONTROL DIGIT 2

     ENTER CONTROL DIGIT
2
      ENTER TRANSIENT RESPONSE OF INTEREST
1
      ENTER INPUT VARIABLE
1
      ENTER INPUT VARIABLE FORCING FUNCTION
1
      ENTER STEP MAGNITUDE
1.0
      LINEAR ROOT CONTROL VECTOR
         2     1     1
      LINEAR ROOT DATA
           A            A0            A1            A2
       -0.7503E 00   0.4718E-03   0.0000E 00   0.0000E 00
        0.0000E 00   0.1000E 00   0.0000E 00   0.0000E 00

      QUADRATIC ROOT CONTROL VECTOR
         1     1
      QUADRATIC ROOT DATA
           A            B            THETA         C           THETA'
       -0.4581E 00   0.6649E-01   0.4643E 01   0.1002E 01
```

FIGURE A-9 CALCULATION OF THE INVERSE
LAPLACE TRANSFORMATION

the user is informed of the number of roots that exist, their type and multiplicity.

An example of the output is given in Figure A-9. The linear root control vector informs the user the number of linear roots and the number of times each is repeated. The coefficients associated with Equation (A-5) are given under the linear root data.

$$F(t) = e^{At} (A0 + A1*t + A2*t*t) + H(t) \tag{A-5}$$

where s = A is a linear root repeated n-1 times

and $$F(s) = \frac{\phi(s)}{(S-A)^n} + H(s), \; n = 1,2,3$$

The presence of quadratic roots result in a similar display of data. If the quadratic factor appears only once in the overall transfer function it takes the form of Equation (A-6) while it assumes the form of Equation (A-7) if it appears repeated once.

$$F(t) = C\ e^{At}\ SIN\ (B*t + THETA) + H(t) \tag{A-6}$$

where s = A + iB is a single quadratic factor

and $$F(s) = \frac{\phi(s)}{(S-A)^2 + B^2} + H(s)$$

$$F(t) = e^{At}\{C(SIN(B*t + THETA)-B*t*COS(B*t + THETA)) + C'*COS(B*t + THETA')\} \qquad (A-7)$$

where S = A + iB is a repeated quadratic factor

$$\text{and } F(s) = \frac{\phi(s)}{[(S-A)^2 + B^2]^2} + H(s)$$

A.4.4. Transient Response Data Calculation

The partial fraction coefficients must have been previously calculated before the transient response can be determined. If any modification to the block diagram transfer functions has taken place since the last partial fraction coefficients were calculated, an error will result when this phase of the program is executed. This error message is shown below and also appears in context in Figure A-9.

```
INVERSE TRANSFORMATION HAS NOT BEEN CALCULATED
SINCE TRANSFER FUNCTION AREA CHANGED
ENTER CONTROL DIGIT 2
```

Once the inverse Laplace transformation with associated partial fraction coefficients is determined, the program then requests the user to supply parameters about the transient response calculation. The transient response replot digit, the total time, and the number of

data values to be calculated are included as the parameters. The last two parameters are only required if an original plot of the transient response data is requested as given by the replot digit. (Figure A-10)

A maximum of five hundred transient response data points can be calculated. The time axis always starts at zero time and proceeds to the total time. The response data is stored into a disk file immediately after it is calculated. Because of this feature the user can display this transient response data at a later time, by means of the replot option, without recalculation.

Figure A-14 gives some of the errors that can occur when entering the parameters necessary for the transient response data calculation and display.

A.4.5. <u>Display of Transient Response Data</u>

After the transient response data has been calculated and stored on disk, information concerning the display of the data is requested. This input includes the display device to be used, scope or plotter, the plot direction, and the title for the display. Four possible graph types, as listed below and shown in Figures A-11 to A-13, are available.

```
      ENTER TIME RESPONSE REPLOT DIGIT
      0 = ORIGINAL , 1 = REPLOT
0
      ENTER TOTAL TIME
15
      ENTER NUMBER OF TIME DIVISIONS, MAX = 500
250
      ENTER PLOT DEVICE
      0 = PLOTTER, 1 = SCOPE
0
      ENTER PLOT DIRECTION
      1 = VERTICAL
      2 = HORIZONTAL
      3 = TOP PLOT
      4 = BOTTOM PLOT
1
      ENTER TITLE OF PLOT - 33 CHAR. MAX.
VERTICAL PLOT FOR USER'S MANUAL

     ENTER CONTROL DIGIT
3
      ENTER TIME RESPONSE REPLOT DIGIT
1
      ENTER PLOT DEVICE
0
      ENTER PLOT DIRECTION
2
      ENTER TITLE OF PLOT - 33 CHAR. MAX.
HORIZONTAL PLOT FOR THE USER'S MANUAL

     ENTER CONTROL DIGIT
3
      ENTER TIME RESPONSE REPLOT DIGIT
1
      ENTER PLOT DEVICE
0
      ENTER PLOT DIRECTION
3
      ENTER TITLE OF PLOT - 33 CHAR. MAX.
TOP HALF OF PAGE PLOT
```

FIGURE A-10 CALCULATION AND DISPLAY OF THE TRANSIENT RESPONSE DATA

```
     ENTER CONTROL DIGIT
1

      ENTER TRANSFER FUNCTION DATA
C1(1.1,1)(2)/(1)

      ENTER TRANSFER FUNCTION DATA

      ILLEGAL TRANSFER FUNCTION NAME

     ENTER CONTROL DIGIT
2
      ENTER TRANSIENT RESPONSE OF INTEREST
1
      ENTER INPUT VARIABLE
4

     ENTER CONTROL DIGIT
3
      ENTER TIME RESPONSE REPLOT DIGIT
0
      ENTER TOTAL TIME
15
      ENTER NUMBER OF TIME DIVISIONS, MAX = 500
250
      ENTER PLOT DEVICE
0
      ENTER PLOT DIRECTION
4
      ENTER TITLE OF PLOT - 33 CHAR. MAX.
BOTTOM HALF OF PAGE PLOT
```

FIGURE A-10 (CONT'D)

i) Vertical Plot - Figure A-11

ii) Horizontal Plot - Figure A-12

iii) Top Half Page Plot - Figure A-13

iv) Bottom Half Page Plot - Figure A-13

The transient response data can also be listed on the line printer. This is accomplished by having data switch 6 set in the "ON" position when this phase of the program is entered. (Table A-5)

A.4.6. Performance Criteria

A number of performance criteria can be calculated from the transient response such as rise time, steady state value, and percent overshoot to mention a few. These performance criteria are closely associated with the transient response of the system and can therefore be displayed with it. If a display of the criteria is requested it appears on the bottom half of a plot of the type shown in Figure A-13 if the results are displayed on the scope or the plotter. Another option allows the user to get a listing of the criteria on his typewriter.

Figure A-14 shows the calculated values for the different performance criteria while Figure A-15 shows this same information as well as a plot of the transient response.

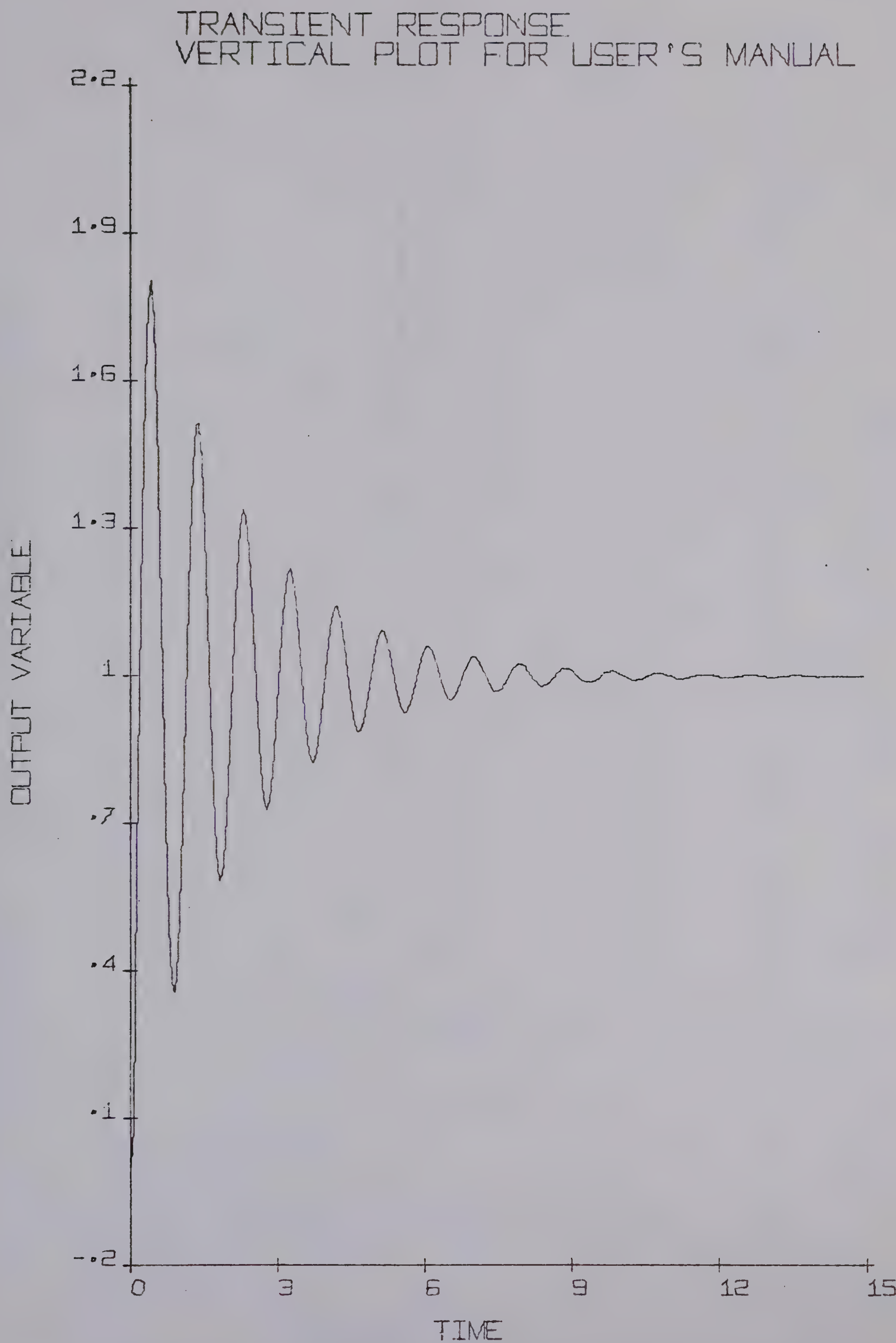


FIGURE A-11 Example of a Vertically Plotted Graph

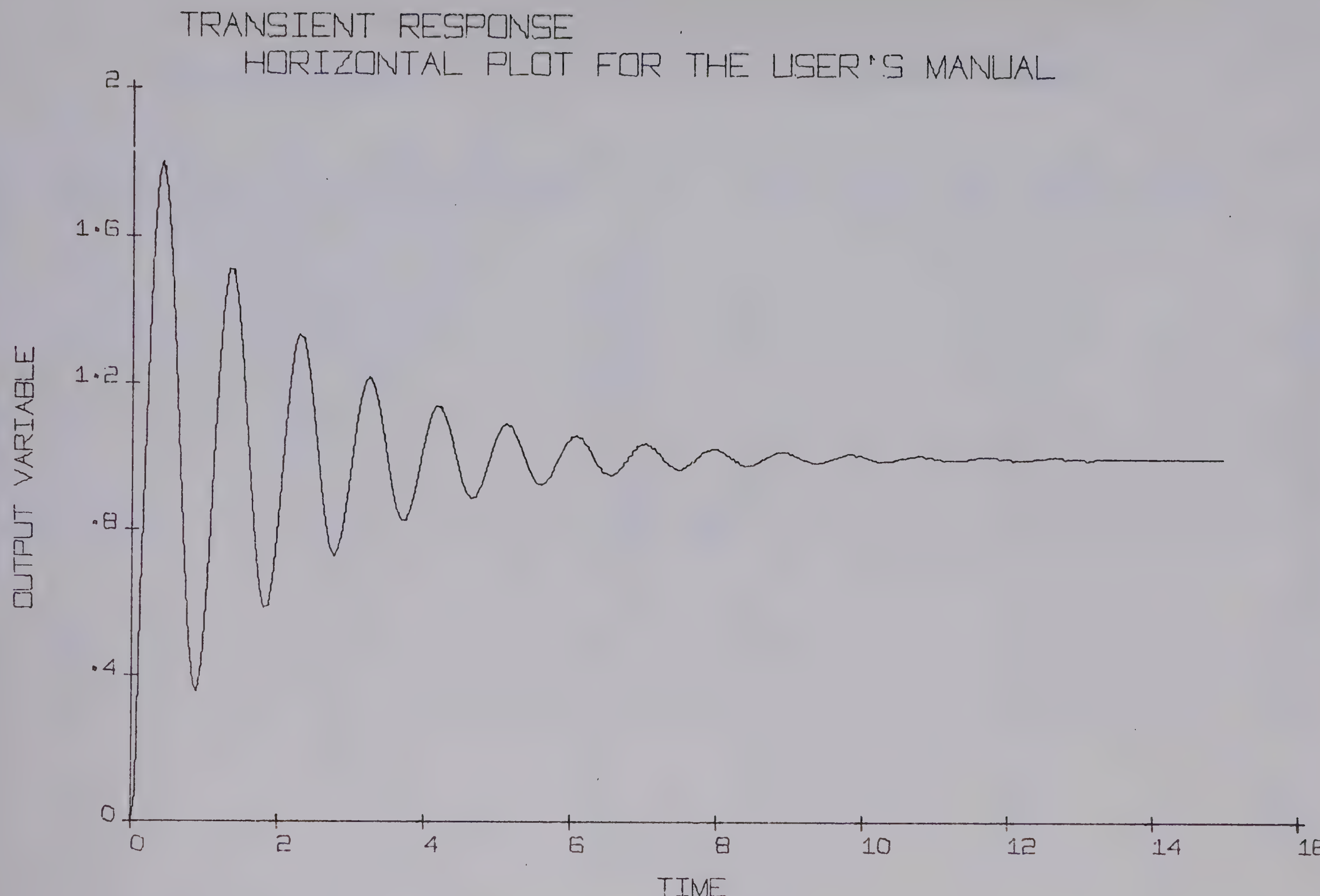
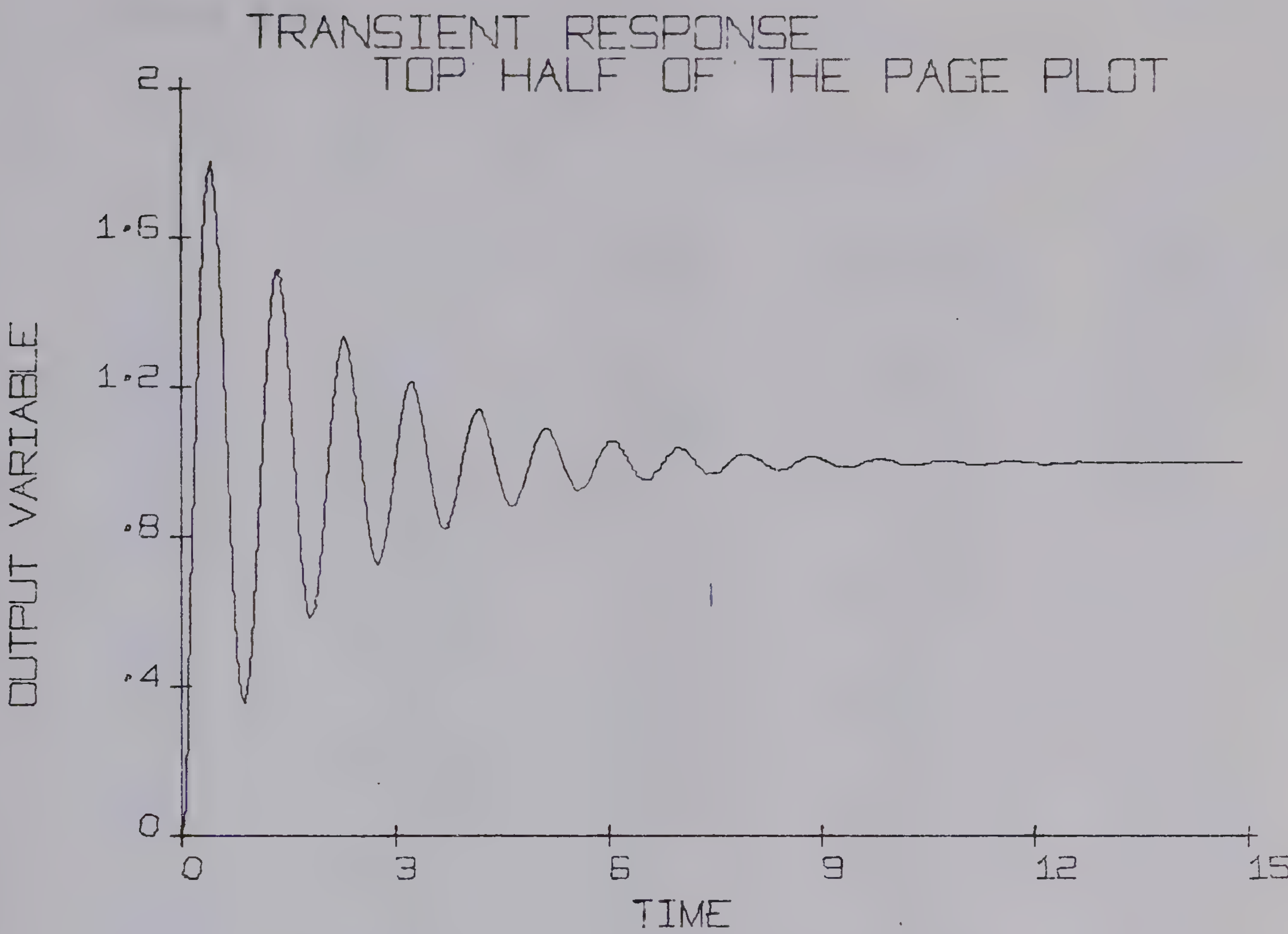


FIGURE A-12 Example of a Horizontally Plotted Graph



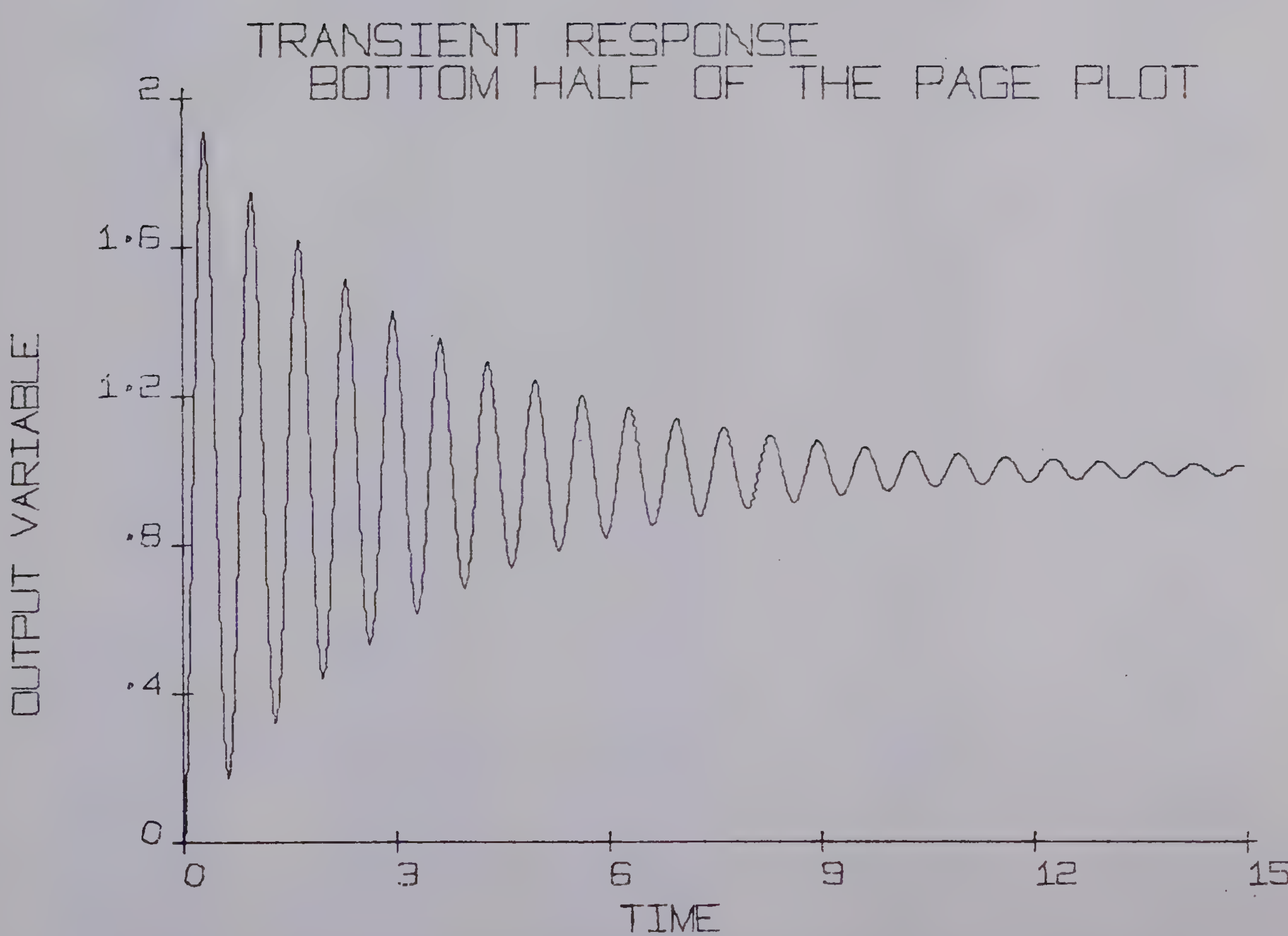


FIGURE A-13   Example of Two Plots Per Graph

TABLE A-5 TRANSIENT RESPONSE DATA

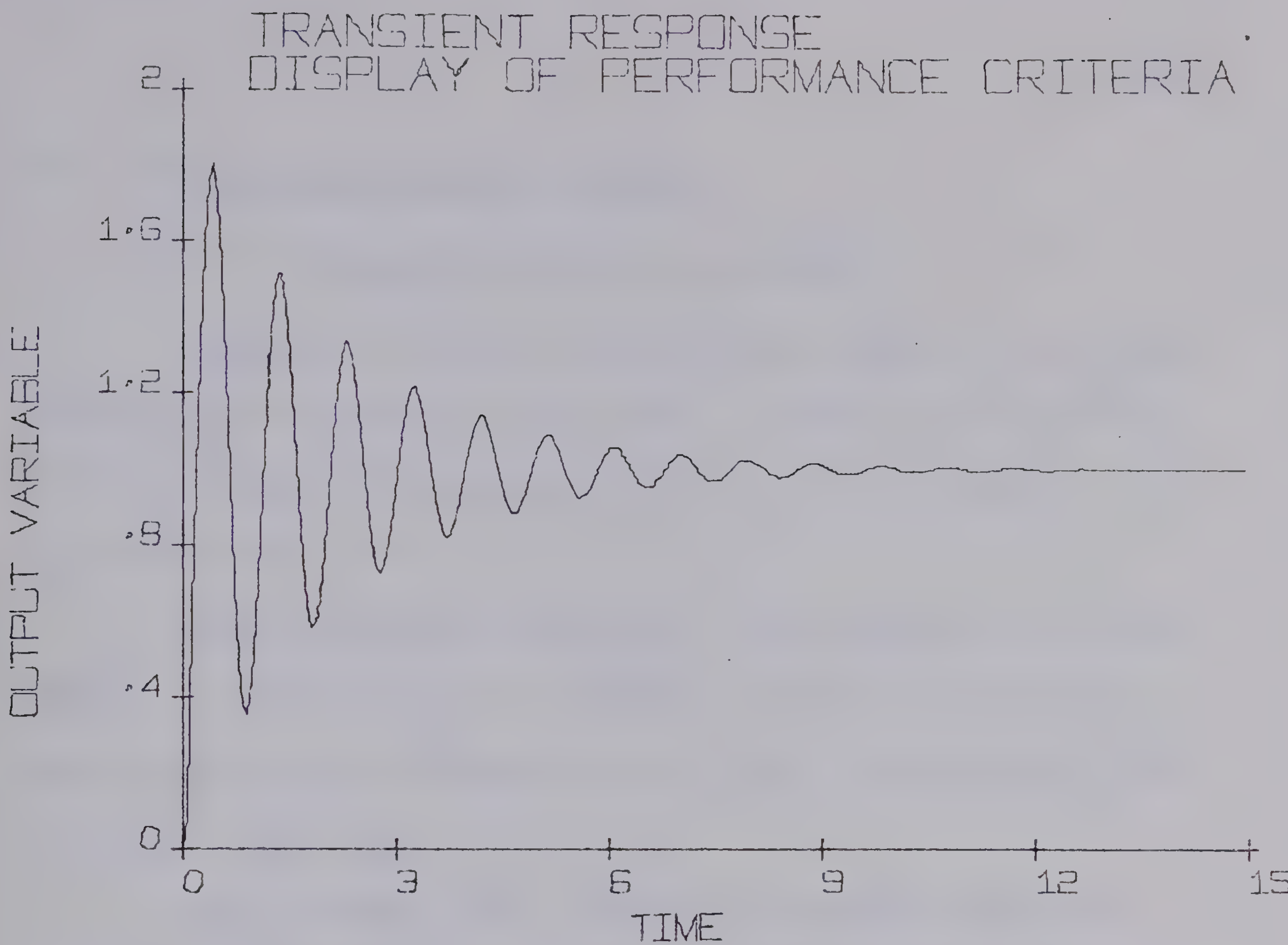
*** TOP HALF PAGE PLOT ***

START =0.0000 TOTAL TIME = 15.00000 STEP SIZE = 0.06024

0.0000	0.0781	0.6089	0.9669	1.3107
1.5881	1.7596	1.8044	1.7224	1.5330
1.2713	0.9818	0.7106	0.4959	0.3760
0.3564	0.4374	0.6012	0.8179	1.0512
1.2640	1.4246	1.5109	1.5135	1.4365
1.2961	1.1175	0.9302	0.7639	0.6430
0.5839	0.5923	0.6634	0.7828	0.9294
1.0790	1.2084	1.2986	1.3377	1.3225
1.2582	1.1574	1.0377	0.9185	0.8184
0.7517	0.7271	0.7460	0.8032	0.8878
0.9852	1.0796	1.1567	1.2054	1.2197
1.1992	1.1490	1.0784	0.9995	0.9249
0.8659	0.8309	0.8236	0.8445	0.8881
0.9465	1.0103	1.0689	1.1138	1.1386
1.1406	1.1208	1.0833	1.0350	0.9838
0.9379	0.9040	0.8868	0.8881	0.9066
0.9385	0.9783	1.0192	1.0550	1.0804
1.0919	1.0886	1.0717	1.0447	1.0122
0.9795	0.9518	0.9330	0.9255	0.9300
0.9451	0.9679	0.9944	1.0203	1.0417
1.0555	1.0599	1.0549	1.0416	1.0226
1.0011	0.9806	0.9641	0.9542	0.9516
0.9570	0.9686	0.9844	1.0018	1.0179
1.0304	1.0376	1.0385	1.0334	1.0234
1.0103	0.9963	0.9837	0.9742	0.9692
0.9693	0.9741	0.9826	0.9934	1.0046
1.0145	1.0216	1.0250	1.0243	1.0199
1.0126	1.0038	0.9949	0.9872	0.9819
0.9797	0.9807	0.9847	0.9908	0.9980
1.0051	1.0111	1.0150	1.0163	1.0151
1.0116	1.0064	1.0006	0.9950	0.9904
0.9876	0.9868	0.9881	0.9912	0.9954
1.0002	1.0046	1.0081	1.0101	1.0105
1.0092	1.0065	1.0030	0.9992	0.9957
0.9930	0.9916	0.9915	0.9928	0.9951
0.9980	1.0010	1.0038	1.0058	1.0068
1.0066	1.0055	1.0035	1.0011	0.9987
0.9966	0.9951	0.9944	0.9947	0.9957
0.9973	0.9993	1.0013	1.0029	1.0040
1.0044	1.0041	1.0032	1.0018	1.0002
0.9987	0.9974	0.9966	0.9964	0.9967
0.9975	0.9986	0.9999	1.0012	1.0021
1.0027	1.0028	1.0025	1.0018	1.0008
0.9998	0.9988	0.9981	0.9977	0.9976
0.9980	0.9986	0.9994	1.0002	1.0010
1.0015	1.0018	1.0018	1.0015	1.0010
1.0003	0.9996	0.9991	0.9986	0.9984
0.9985	0.9988	0.9992	0.9997	1.0003
1.0007	1.0010	1.0012	1.0011	1.0009
1.0005	1.0000	0.9996	0.9993	0.9990

```
     ENTER CONTROL DIGIT
3
      ENTER TIME RESPONSE REPLOT DIGIT
3
      INPUT IS IN ERROR - TRY AGAIN
1
      ENTER PLOT DEVICE
2
      INPUT IS IN ERROR - TRY AGAIN
0
      ENTER PLOT DIRECTION
3
      ENTER TITLE OF PLOT - 33 CHAR. MAX
DISPLAY OF PERFORMANCE DATA

     ENTER CONTROL DIGIT
4
      ENTER DEVICE FOR PERFORMANCE DATA OUTPUT
      0 = PLOTTER, 1 = SCOPE, 2 = TYPEWRITER
3
      INPUT IS IN ERROR - TRY AGAIN
2


      PERFORMANCE CRITERIA FOR ABOVE RESPONSE
      STEADY STATE          =     0.9996
      PERCENT OVERSHOOT     =      80.50
      PEAK TIME             =     0.4216
      RISE TIME  0 TO 100 =       0.1863
      RISE TIME 10 TO  90 =       0.0944
      DELAY TIME            =     0.1578
      DECAY RATIO           =     0.6384
      SETTLING TIME         =     6.1445
      ERROR INTEGRALS
           IAE  =      0.05786
           ISE  =      0.05568
           ITAE =      0.02775

     ENTER CONTROL DIGIT
4
      ENTER DEVICE FOR PERFORMANCE DATA OUTPUT
      0 = PLOTTER, 1 = SCOPE, 2 = TYPEWRITER
0

     ENTER CONTROL DIGIT
A
      INPUT IS IN ERROR - TRY AGAIN
9
```

FIGURE A-14 CALCULATION AND DISPLAY OF THE PERFORMANCE CRITERIA



PERFORMANCE CRITERIA FOR ABOVE RESPONSE

STEADY STATE	=	0.9995
PERCENT OVERSHOOT	=	80.50
PEAK TIME	=	0.4216
RISE TIME 0 TO 100	=	0.1863
RISE TIME 10 TO 90	=	0.0944
DELAY TIME	=	0.1578
DECAY RATIO	=	0.6384
SETTLING TIME	=	6.1445

ERROR INTEGRALS

IAE	=	0.05786
ISE	=	0.05568
ITAE	=	0.02775

FIGURE A-15 Transient Response Plus Performance Criteria Display

A.5. Frequency Response Methods

A.5.1. Frequency Response Options

Frequency response methods are based on the open-loop transfer function G(s) H(s) as given in Figure A-1. Plots are made by replacing s by $j\omega$ in the transfer function, G(s) H(s).

Three methods of plotting this function are most useful in the frequency response analysis and design of feedback control systems and will be illustrated in the following sections.

Upon entering the frequency response analysis section of the CSDAP system, some general parameters must be entered by the user. These include the frequency response replot digit and the type of frequency response diagram desired, that is Bode, Nyquist, or Log-Modulus. If a replot of data existing on disk has been requested, the CSDAP system proceeds directly to the phase pertaining to the type of plot desired. If a replot was not requested, the user is asked to enter the initial and terminal frequencies for the range of interest and the number of points desired. The points are distributed evenly over the complete frequency range. The program then proceeds to the correct phase to display the frequency response diagram desired.

A.5.2. Bode Plot

The Bode plot, which is also known as a Corner plot, is composed of two separate graphs. The first graph is a semilog plot of the magnitude, in decibels, versus the frequency, in radians per second, on a logarithmic scale. The second graph is also a semilog plot with the phase angle, in degrees, being displayed versus the frequency on a logarithmic scale.

The user needs only to specify the device to be used to display the plot and the desired title for the plot in order to obtain a Bode plot. Figure A-16 shows typical user supplied information while Figure A-17 is an example of a Bode plot.

The gain margin, gain crossover frequency, phase margin, and phase crossover frequency values are given in the lower left hand corner of the two plots.

A.5.3. Nyquist Plot

The Nyquist plot sometimes known as the Polar Plot displays the magnitude in decibels versus the phase angle in degrees on rectangular coordinates as the frequency is varied from the initial to the terminal value as specified by the user.

The user can specify the scaling of the plot or use automatic scaling. The plot device must be selected

and if desired a title for the Nyquist plot to be shown. Specifying automatic scaling ensures that the axes will be labelled large enough to accommodate all data values. Figure A-18 is an example of a Nyquist plot with automatic scaling. A user specified scale allows the display of a small section of the plot in more detail such as shown in Figure A-19. The user supplied information to obtain a Nyquist plot is shown in Figure A-16 for both types of scaling.

A.5.4. Log-Modulus Plot

The log-modulus or gain phase plot displays the magnitude in decibels versus the phase angle of $G(j\omega)\ H(j\omega)$ in degrees in rectangular coordinates with frequency as the parameter on the curve. The plot is a continuous curve plot with each value shown by a small plus sign as can be seen in Figure A-20.

In addition to the log-modulus curve, some frequency response specifications are also listed at the bottom of the graph. These include the peak resonance and resonant frequency in addition to the gain and phase margins and phase crossover points that can also be displayed with the Bode plot.

As can be seen from Figure A-16, for the log-modulus plot, the user selects the plot device and then specifies the appropriate title.

```
     ENTER CONTROL DIGIT
5
      ENTER FREQUENCY RESPONSE REPLOT DIGIT
      0 = ORIGINAL , 1 = REPLOT
0
      ENTER FREQUENCY RESPONSE TYPE
      0=NONE, 1=BODE, 2=NYQUIST, 3=LOG-MODULUS
1
      ENTER INITIAL AND TERMINAL FREQUENCIES
0.01 100
      ENTER NUMBER OF POINTS - 100 MAX
100
      ENTER PLOT DEVICE
      0 = PLOTTER, 1 = SCOPE
0
      ENTER TITLE OF BODE PLOT , 36 CHAR.
EXAMPLE FOR THE USER'S MANUAL

     ENTER CONTROL DIGIT
5
      ENTER FREQUENCY RESPONSE REPLOT DIGIT
1
      ENTER FREQUENCY RESPONSE TYPE
2
      ENTER TYPE OF SCALE
      0 = AUTOMATIC , 1 = USER SPECIFIED
0
      ENTER PLOT DEVICE
      0 = PLOTTER, 1 = SCOPE
0
      ENTER TITLE OF NYQUIST PLOT , 36 CHAR.
EXAMPLE OF AUTOMATIC SCALING
```

FIGURE A-16 INPUT DATA FOR FREQUENCY RESPONSE CALCULATIONS

```
     ENTER CONTROL DIGIT
5
      ENTER FREQUENCY RESPONSE REPLOT DIGIT
1
      ENTER FREQUENCY RESPONSE TYPE
2
      ENTER TYPE OF SCALE
      0 = AUTOMATIC , 1 = USER SPECIFIED
1
      ENTER REAL INITIAL AND TERMINAL FREQUENCIES
-40 .0 0.0
      ENTER IMAGINARY INITIAL AND TERMINAL FREQUENCIES
-60.0 0.0
      ENTER PLOT DEVICE
      0 = PLOTTER, 1 = SCOPE
1

      ENTER TITLE OF NYQUIST PLOT , 36 CHAR.
EXAMPLE OF USER SPECIFIED SCALE


     ENTER CONTROL DIGIT
5
      ENTER FREQUENCY RESPONSE REPLOT DIGIT
      0 = ORIGINAL , 1 = REPLOT
0
      ENTER FREQUENCY RESPONSE TYPE
3
      0=NONE, 1=BODE, 2=NYQUIST, 3=LOG-MODULUS
      ENTER INITIAL AND TERMINAL FREQUENCIES
0.1 50.0
      ENTER NUMBER OF POINTS - 100 MAX
50
      ENTER TITLE FOR FREQUENCY RESPONSE DATA
LOG-MODULUS DATA DUMP
      ENTER PLOT DEVICE
      0 = PLOTTER, 1 = SCOPE
0

      ENTER TITLE OF LOG-MODULUS PLOT, 36 CHAR.
EXAMPLE FOR USER'S MANUAL
     ENTER CONTROL DIGIT
9
```

FIGURE A-16 (CONT'D)

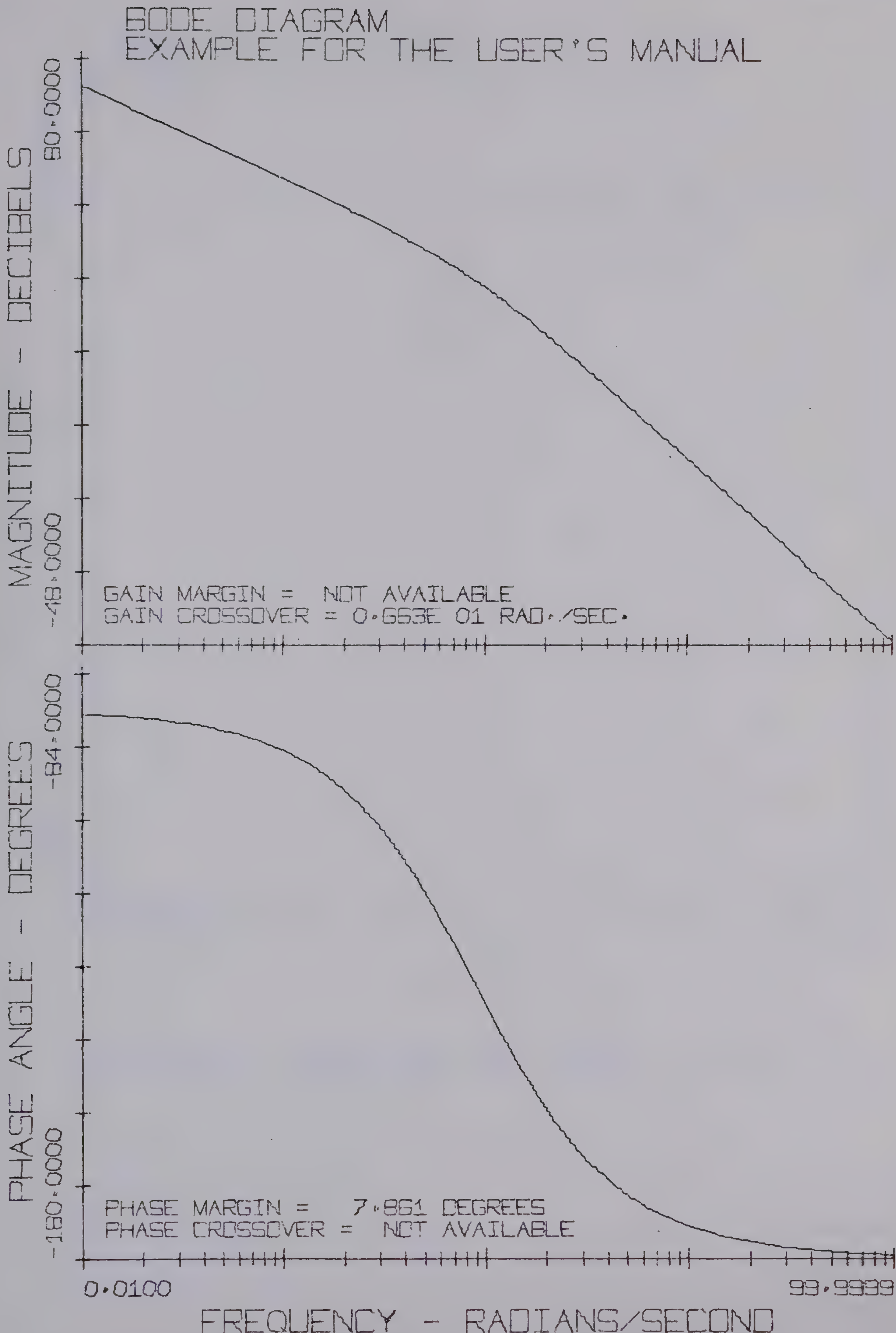


FIGURE A-17 Example of a Bode Plot

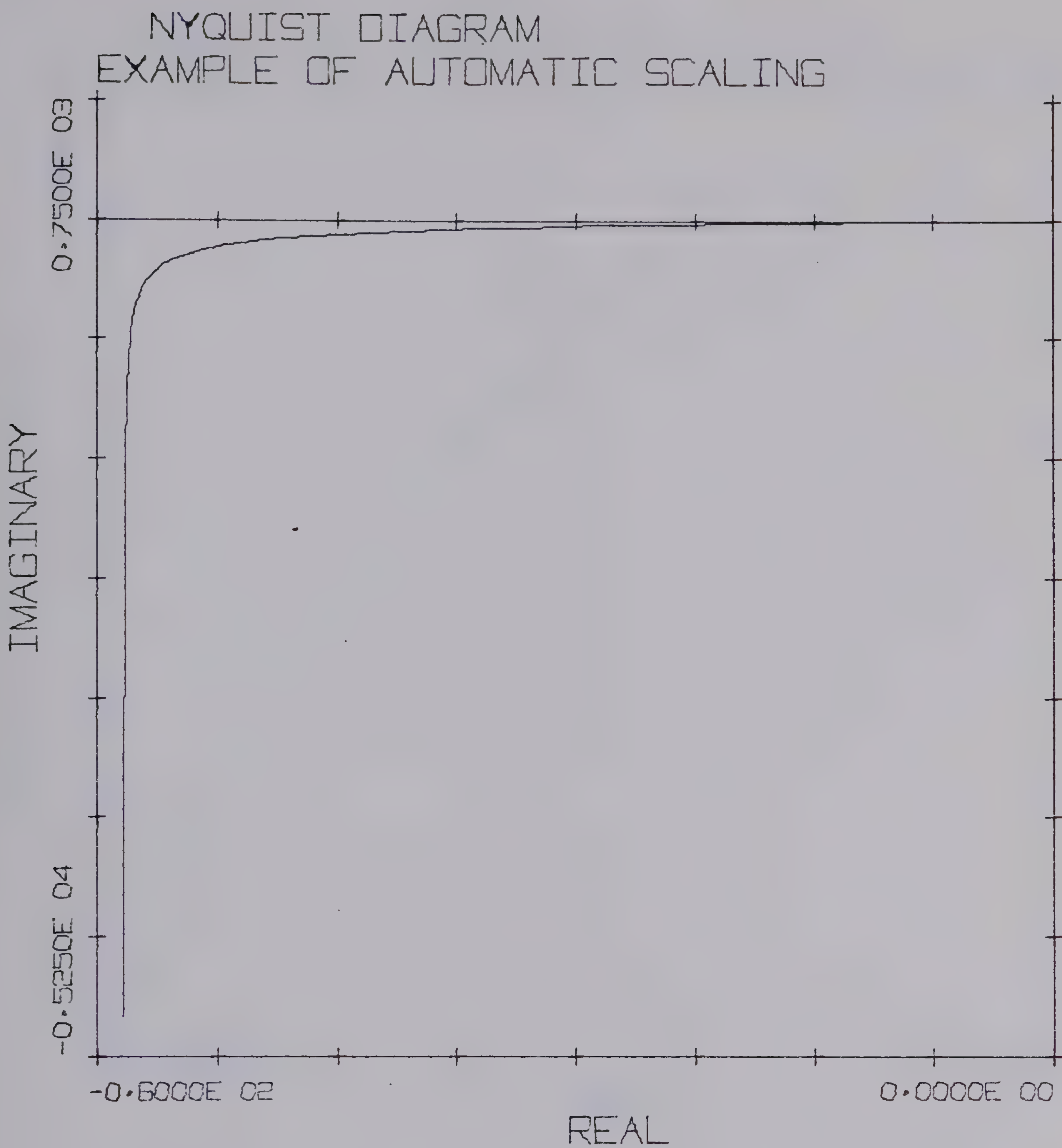


FIGURE A-18 Nyquist Plot with Automatic Scaling

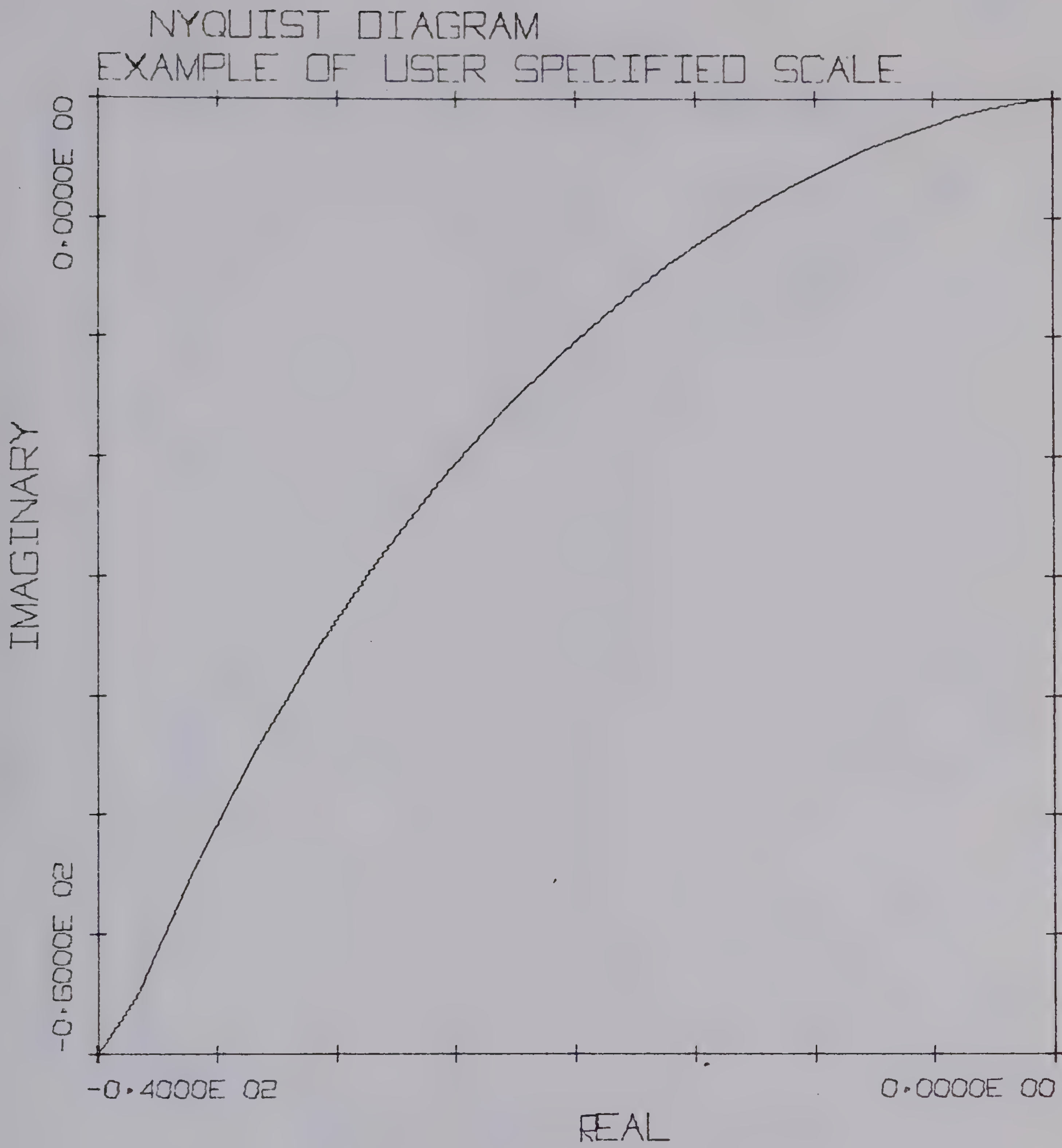
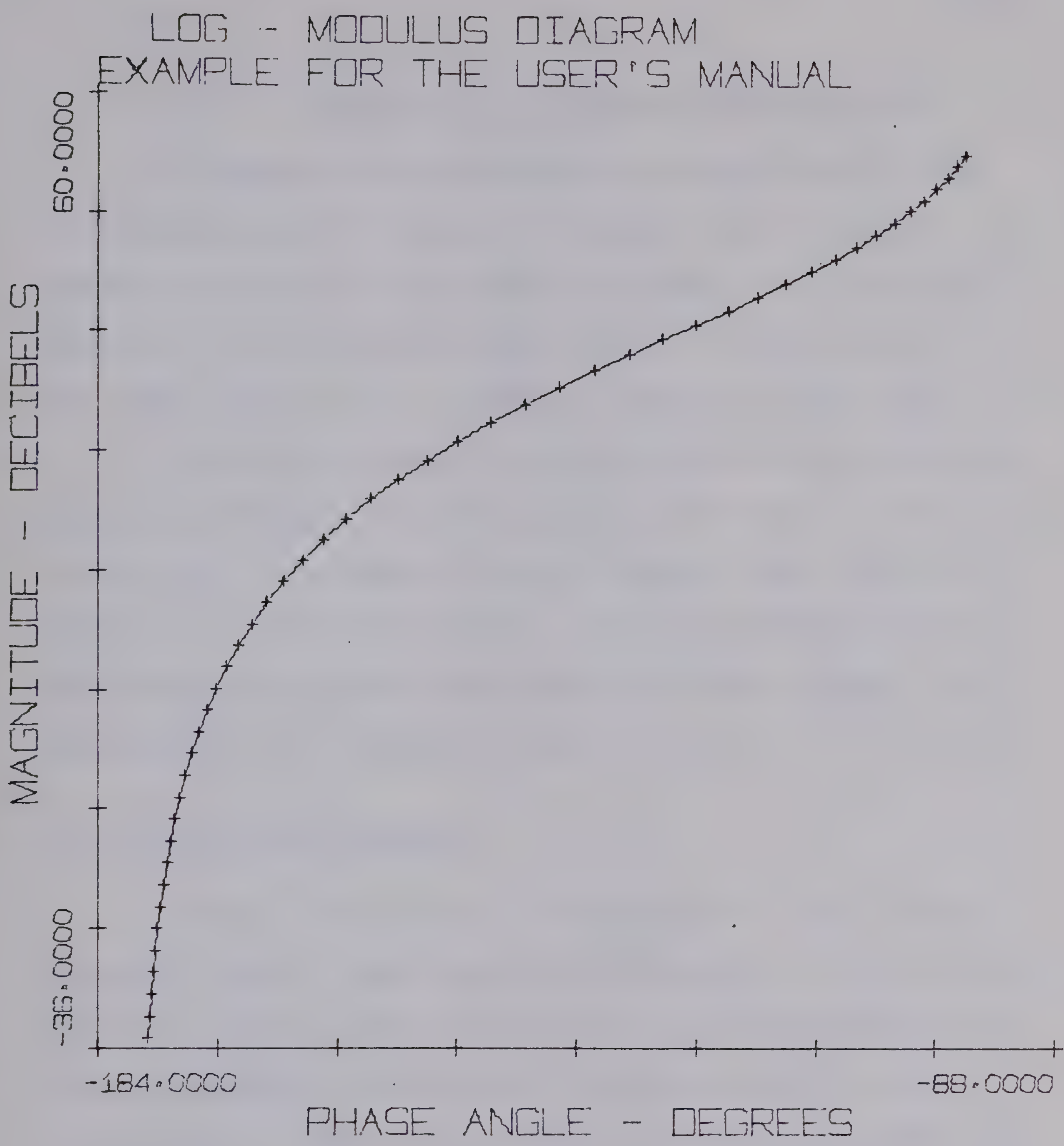


FIGURE A-19 Nyquist Plot with User Specified Scaling



PERFORMANCE CRITERIA FOR ABOVE DIAGRAM

PHASE CROSSOVER	=	NOT AVAILABLE
GAIN MARGIN	=	NOT AVAILABLE
GAIN CROSSOVER	=	6.6337 RAD./SEC.
PHASE MARGIN	=	7.8611 DEGREES
PEAK RESONANCE	=	7.2631 DECIBELS
RESONANT FREQUENCY	=	6.5053 RAD./SEC.

FIGURE A-20 Example of a Log-Modulus Plot

A.5.5. Tabulation of Frequency Response Data

A listing of the frequency response data on the line printer can be obtained whenever the frequency response section of the CSDAP is entered. This optional listing is controlled by data switch 10 which is set to the "ON" position if a listing is desired.(Table A-6)

The listing option is independent of the particular frequency response plot desired and the replot option. If a Bode plot is desired, then the program uses the same title as for the Bode diagram. If the requested plot of the frequency response data was not a Bode diagram, then the program will request an appropriate title. (Table A-7)

A.6. Root Locus Analysis

The root locus data is displayed on a plot device as a point plot. Both positive and negative values of the gain are shown on this plot. They are differentiated by plotting frequency points with positive gains as small plus signs (+) while those points with negative gains are plotted as small crosses (x).

The user must specify the replot option digit (same as for time response or frequency response), the initial and terminal frequency values, the plot device and the title of the plot (Figure A-22) as shown in Figure A-21.

TABLE A-6 FREQUENCY RESPONSE DATA

*** EXAMPLE FOR THE USER'S MANUAL *** PAGE 1 OF 2

FREQUENCY (RADIANS/SECOND)	MAGNITUDE (DECIBELS)	PHASE ANGLE (DEGREES)
0.01000000	73.97010314	-90.66915273
0.01098498	73.16264748	-90.73434400
0.01206503	72.35513019	-90.80580997
0.01322940	71.54749000	-90.88430404
0.01451827	70.73977279	-90.97042608
0.01593281	69.93193269	-91.06496047
0.01749526	69.12403094	-91.1[illegible]
0.01921907	68.31549061	-91.28249549
0.02110590	67.50792775	-91.40737247
0.02311125	66.69952619	-91.54446029
0.02538659	65.89103622	-91.69460592
0.02788553	65.08247709	-91.85908235
0.03054847	64.27365267	-92.04098465
0.03352992	63.46481296	-92.23964977
0.03671369	62.65529044	-92.45766162
0.04038005	61.84560456	-92.69874255
0.04431605	61.03565597	-92.95906114
0.04863582	60.22519367	-93.24572412
0.05337677	59.41419315	-93.56222029
0.05857997	58.60255438	-93.90817213
0.06428044	57.79016202	-94.28743219
0.07055771	56.97584693	-94.70314025
0.07743599	56.16243994	-95.15804305
0.08498493	55.34671807	-95.65701892
0.09326992	54.52943521	-96.20400142
0.10236253	53.71027612	-96.80196523
0.11234173	52.88887941	-97.45613904
0.12329395	52.06474907	-98.17129039
0.13531294	51.23754299	-98.95247588
0.14850592	50.40548629	-99.80507143
0.16298402	49.57094293	-100.73451028
0.17887376	48.73003342	-101.74847008
0.19631258	47.88247395	-102.84854100
0.21545181	47.02421943	-104.04061555
0.23645707	46.15502350	-105.33370008
0.25950136	45.29163637	-106.73088355
0.28481124	44.40652674	-108.23137413
0.31257693	43.50787701	-109.85356163
0.34305092	42.59301914	-111.58464550
0.37650028	41.66217851	-113.45004853
0.41320740	40.71079897	-115.33984056
0.45349155	39.73738920	-117.45995324
0.49770751	38.73967295	-119.63484239
0.54623256	37.71559729	-121.90804910
0.59948814	36.66244763	-124.26951893
0.65793663	35.57487077	-126.70598530
0.72208330	34.46299283	-129.20401668
0.79248455	33.31345337	-131.74656241
0.86974929	32.12931370	-134.31659054
0.95454755	30.91007545	-136.89419245

TABLE A-6 FREQUENCY RESPONSE DATA

*** EXAMPLE FOR THE USER'S MANUAL *** PAGE 2 OF 2

FREQUENCY (RADIANS/SECOND)	MAGNITUDE (DECIBELS)	PHASE ANGLE (DEGREES)
1.04761366	29.65572705	-139.46016812
1.14975383	28.36672994	-141.99523210
1.26185163	27.04409906	-144.48066633
1.38487976	25.68921086	-146.90025186
1.51990202	24.30368391	-149.23834252
1.6[illegible]	22.[illegible]	-151.4[illegible]
1.83072451	21.45046439	-153.62358474
2.00921673	19.98707395	-155.65375028
2.20511123	18.50248059	-157.56874346
2.42010593	16.99399646	-159.36654114
2.65608160	15.47895774	-161.04702854
2.91502319	13.94445317	-162.61205053
3.19923172	12.39742438	-164.06474494
3.51115164	10.83960920	-165.40950687
3.85347557	9.27258394	-166.65069774
4.22916748	7.69769431	-167.79461007
4.64152061	6.11613307	-168.84697514
5.09406613	4.52893161	-169.81323790
5.59072995	2.93898816	-170.69946455
6.13581810	1.34104354	-171.51144146
6.73404818	-0.25824383	-172.25467491
7.39060960	-1.86034579	-172.93448543
8.11118008	-3.46473703	-173.55591750
8.90200923	-5.07119677	-174.12364625
9.76993876	-6.67523765	-174.64207029
10.72249475	-8.28865301	-175.11531090
11.76791836	-9.89919297	-175.54718232
12.91527548	-11.51069025	-175.94122123
14.17449325	-13.12296803	-176.30059623
15.55648681	-14.73590526	-176.62835216
17.07321521	-16.34937480	-176.92716503
18.73782369	-17.96330964	-177.19984909
20.56473374	-19.57762128	-177.44804954
22.56977418	-21.19224819	-177.67445826
24.77028477	-22.80712884	-177.88084363
27.18535459	-24.42222866	-178.06892600
29.83587977	-26.03750532	-178.24034166
32.74484026	-27.65293195	-178.39659166
35.93740761	-29.26[illegible]4815	-178.53896975
39.44124466	-30.88413119	-178.66870065
43.28670173	-32.49937685	-178.78692985
47.50714499	-34.11569555	-178.89465403
52.13895399	-35.73150059	-178.99283313
57.22425621	-37.34748697	-179.08230781
62.80155009	-38.96343548	-179.16384675
68.92461228	-40.57943212	-179.23812099
75.64460802	-42.19543546	-179.30576372
83.01985406	-43.81149256	-179.36745530
91.11417505	-45.42755740	-179.42355560
99.99767756	-47.04362994	-179.47467044

TABLE A-7 FREQUENCY RESPONSE DATA

*** LOG-MODULUS DATA DUMP *** PAGE 1 OF 1

FREQUENCY (RADIANS/SECOND)	MAGNITUDE (DECIBELS)	PHASE ANGLE (DEGREES)
0.10001000	53.91520816	-96.64742231
0.11353225	52.79563689	-97.53401803
0.12888303	51.67098295	-98.53506374
0.14630952	50.53958516	-99.66347789
0.16609248	49.40062123	-100.93515217
0.18855057	48.25109153	-102.35835981
0.21404548	47.08871328	-103.95[illegible]
0.24298786	45.91035610	-105.75107254
0.27584382	44.71233689	-107.70620059
0.31314273	43.49034965	-109.88604187
0.35548532	42.23953455	-112.27754351
0.40355351	40.95405515	-114.88112068
0.45812159	39.62972891	-117.65104436
0.52006155	38.25525034	-120.69705409
0.59039222	36.83705214	-123.87556066
0.67022512	35.35952103	-127.19648036
.76085284	33.82057369	-130.62912150
0.86373600	32.21885451	-134.12444591
0.98053086	30.55156168	-137.63605455
1.11311952	28.81926263	-141.11702557
1.26363875	27.02375271	-144.51817750
1.43450735	25.16841063	-147.79553823
1.62847308	23.25793549	-150.91261649
1.84868951	21.29702642	-153.84106567
2.09867221	19.29544125	-156.56492948
2.38245781	17.25559094	-159.07170854
2.70461867	15.18121191	-161.35607578
3.07034181	13.08265791	-163.43023852
3.48551880	10.96265415	-165.30727076
3.95693756	8.82526642	-166.97505558
4.49188952	6.67390885	-168.45646616
5.09929313	4.51141304	-169.82344437
5.78882796	2.34000845	-171.01174926
6.57160024	0.16180945	-172.06557518
7.46023294	-2.02195534	-172.99972071
8.46902295	-4.21001283	-173.82575831
9.61422313	-6.40142773	-174.55585598
10.91427969	-8.59547050	-175.20000479
12.39013284	-10.79155346	-175.75983057
14.06555269	-12.98923047	-176.27200224
15.96752734	-15.18814973	-176.71490502
18.12669545	-17.38803359	-177.10543751
20.57782891	-19.58866721	-177.44964909
23.36040539	-21.78987755	-177.75507536
26.51925054	-23.99154507	-178.02045369
30.10524570	-26.19356310	-178.25605958
34.17615025	-28.39585378	-178.46370530
38.79753610	-30.59835594	-178.64662241
44.04382216	-32.80102723	-178.80779361
49.99953085	-35.00382924	-178.94974136

A listing of the data values including the frequency coordinates and gain values can be obtained on the line printer as shown in Table A-8 by having data switch 14 set in the "ON" position.

```
     ENTER CONTROL DIGIT
6
      ENTER ROOT LOCUS REPLOT DIGIT
      0 = ORIGINAL PLOT, 1 = REPLOT
0
      ENTER INITIAL AND TERMINAL FREQUENCIES
-2.0 0.0
      ENTER PLOT DEVICE
      0 = PLOTTER, 1 = SCOPE
1
      ENTER ROOT LOCUS TITLE - 32 CHAR. MAX.
TITLE FOR THE SCOPE

     ENTER CONTROL DIGIT
6
      ENTER ROOT LOCUS REPLOT DIGIT
      0 = ORIGINAL PLOT, 1 = REPLOT
0
      ENTER INITIAL AND TERMINAL FREQUENCIES
-0.75 -0.25
      ENTER PLOT DEVICE
      0 = PLOTTER, 1 = SCOPE
1
      ENTER ROOT LOCUS TITLE - 32 CHAR. MAX.
TITLE FOR THE SCOPE

     ENTER CONTROL DIGIT
6
      ENTER ROOT LOCUS REPLOT DIGIT
      0 = ORIGINAL PLOT, 1 = REPLOT
0
      ENTER INITIAL AND TERMINAL FREQUENCIES
-1.0 0.0
      ENTER PLOT DEVICE
      0 = PLOTTER, 1 = SCOPE
0
      ENTER ROOT LOCUS TITLE - 32 CHAR. MAX.
EXAMPLE FOR THE USER'S MANUAL
     ENTER CONTROL DIGIT
9
```

FIGURE A-21 INPUT DATA FOR ROOT LOCUS CALCULATION

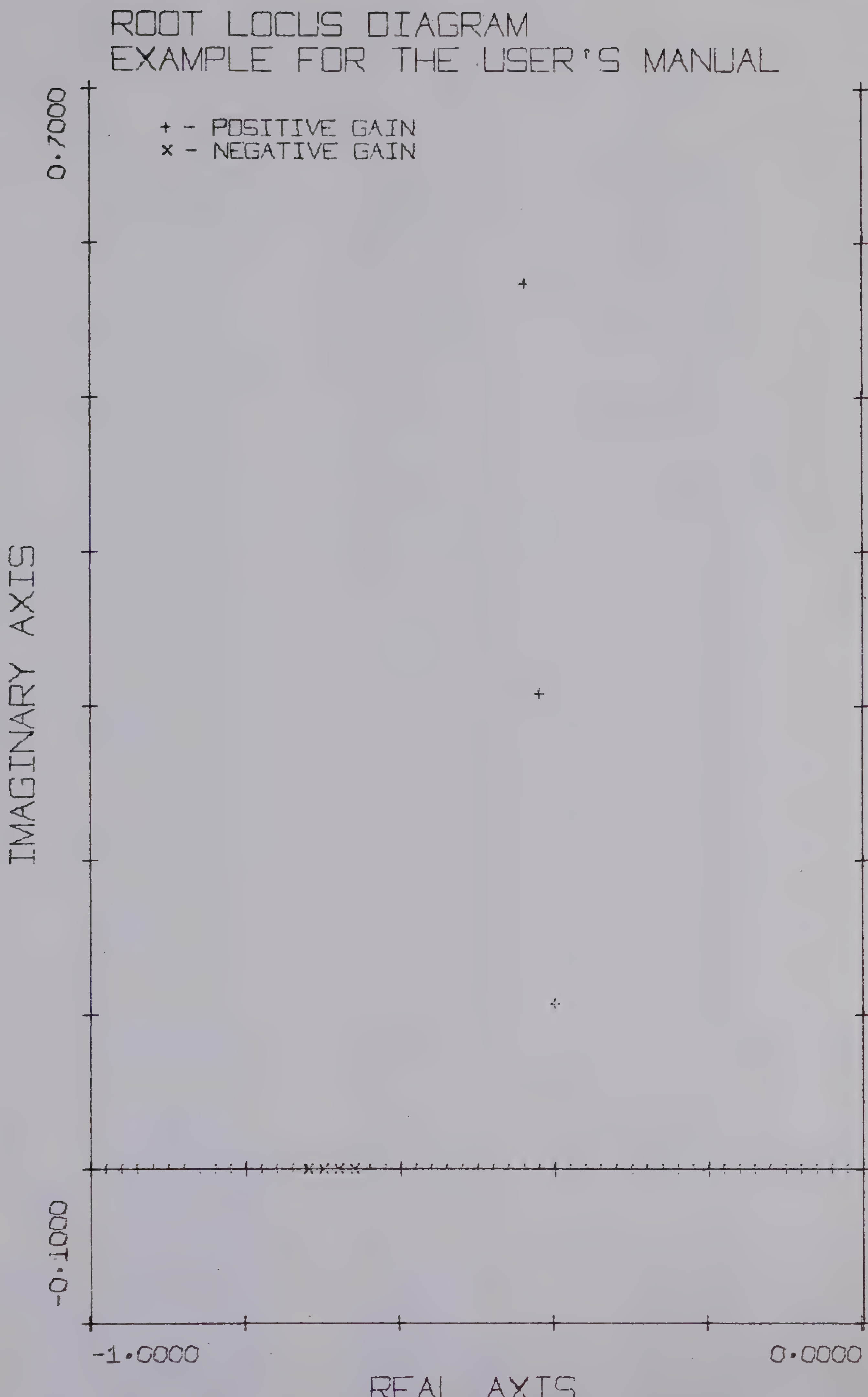


FIGURE A-22 Display of Root Locus

TABLE A-8 ROOT LOCUS POINT DATA

*** EXAMPLE FOR THE USER'S MANUAL *** PAGE 1 OF 2

REAL PART	IMAG. PART	GAIN
-0.4399975	0.5744345	0.0123563
-0.4199975	0.3080416	0.0067279
-0.3999974	0.1008701	0.0044569
-1.0000002	0.0000000	0.0003905
-0.9800000	0.0000000	0.0000000
-0.9599999	0.0000000	0.0006021
-0.9399999	0.0000000	0.0012098
-0.9199997	0.0000000	0.0018261
-0.8999996	0.0000000	0.0024684
-0.8799995	0.0000000	0.0031496
-0.8599995	0.0000000	0.0038991
-0.8399993	0.0000000	0.0047629
-0.8199992	0.0000000	0.0058246
-0.7999992	0.0000000	0.0072749
-0.7799990	0.0000000	0.0096[illegible]
-0.7599989	0.0000000	0.0145932
-0.7399988	0.0000000	0.0383557
-0.7199988	0.0000000	-0.0316854
-0.6999986	0.0000000	-0.0080591
-0.6799985	0.0000000	-0.0031455
-0.6599984	0.0000000	-0.0009213
-0.6399984	0.0000000	0.0003724
-0.6199982	0.0000000	0.0012576
-0.5999981	0.0000000	0.0019024
-0.5799981	0.0000000	0.0024002
-0.5599979	0.0000000	0.0027944
-0.5399978	0.0000000	0.0031125
-0.5199977	0.0000000	0.0033712
-0.4999976	0.0000000	0.0035813
-0.4799975	0.0000000	0.0037500
-0.4599975	0.0000000	0.0038826
-0.4399975	0.0000000	0.0039829
-0.4199975	0.0000000	0.0040537
-0.3999974	0.0000000	0.0040969
-0.3799973	0.0000000	0.0041142
-0.3599973	0.0000000	0.0041070
-0.3399972	0.0000000	0.0040763
-0.3199972	0.0000000	0.0040227
-0.2999972	0.0000000	0.0039471
-0.2799972	0.0000000	0.0038499
-0.2599971	0.0000000	0.0037317
-0.2399971	0.0000000	0.0035927
-0.2199971	0.0000000	0.0034333
-0.1999970	0.0000000	0.0032538
-0.1799970	0.0000000	0.0030545
-0.1599970	0.0000000	0.0028354
-0.1399970	0.0000000	0.0025968
-0.1199970	0.0000000	0.0023368
-0.0999970	0.0000000	0.0020615
-0.0799969	0.0000000	0.0017653

TABLE A-8 ROOT LOCUS POINT DATA

*** EXAMPLE FOR THE USER'S MANUAL *** PAGE 2 OF 2

REAL PART	IMAG. PART	GAIN
-0.0599969	0.0000000	0.0014499
-0.0399969	0.0000000	0.0011156
-0.0199969	0.0000000	0.0007525

A.7. Program Limitations

In the design of the CSDAP system a number of restrictions were imposed on the program due to core size limitations and arbitrarily chosen criteria.

Transfer function data can be entered for any block given in Figure A-3. The symbols used for this data entry must be the same as those given within the blocks of Figure A-3. The CSDAP system can contain a maximum of two hundred coefficients including both those of the numerator and denominator of all the transfer functions listed in Table A-3.

In the calculation of the transient response of the control system, only arbitrary disturbances such as a step, pulse, sine wave, ramp, or impulse may be used as the forcing function. The characteristic equation denominator is restricted to containing no more than twelve linear roots and eight quadratic roots. Each linear root may be repeated twice while a quadratic root may appear in the system repeated only once (two equal roots). When the above rule is violated a suitable error message will be outputted to the user indicating that a linear or quadratic root was repeated n times. It is usually sufficient to solve this problem to reduce the order of the Padé approximation.

The plotting programs in CSDAP only allow one set of calculated values to be plotted on a single graph. This restriction of no multiple plots is viewed as a minor inconvenience because the CSDAP was designed

as an interactive system giving the user a quick view of the resultant plot. Enough other numerical data is given for a more exact comparison of various results.

The Bode diagrams produced by CSDAP are restricted to a phase angle in the angle between -720 and +720 degrees. If the phase angle should lie outside this range, a discontinuity of one cycle of 360 degrees will appear on the diagram.

The CSDAP system does not allow execution to be continued after it is stopped. Each time CSDAP is executed it initializes the transfer function area and all other system parameters. Replots of transient, frequency, and root locus response data that is stored on the disk may be obtained without re-entering the transfer function data.

APPENDIX B

APPENDIX B

USER-PROGRAM CORRESPONDENCE FOR EXAMPLES GIVEN IN CHAPTER VII

CONTROL DIGIT FUNCTIONS

0 = LIST CONTROL DIGIT AND DATA SWITCH OPTIONS
1 = INPUT TRANSFER FUNCTION DATA
2 = CALCULATE INVERSE LAPLACE TRANSFORM
3 = PLOT THE TIME RESPONSE DATA
4 = CALCULATE THE PERFORMANCE CRITERIA
5 = FREQUENCY RESPONSE CALCULATIONS
6 = ROOT LOCUS PLOT
7 = LIST TRANSFER FUNCTION AREA
8 = INITIALIZE TRANSFER FUNCTION AREA
9 = CALL EXIT

DATA SWITCH FUNCTIONS

1 = LIST CONTROL DIGITS AND SWITCH OPTIONS
2 = LIST TRANSFER FUNCTION DATA AFTER INPUT
3 = INPUT-OUTPUT VARIABLE AND FORCING FUNCTION SELECTION
4 = LIST PARTIAL FRACTION DATA
5 = REPLOT, AND PLOT OPTIONS FOR TIME RESPONSE
6 = LIST TIME RESPONSE DATA ON THE PRINTER
7 = PERFORMANCE DATA OUTPUT DEVICE OPTIONS
8 = FREQUENCY RESPONSE DATA OPTIONS
9 = BODE PLOT OPTIONS
10 = LIST FREQUENCY RESPONSE DATA ON THE PRINTER
11 = NYQUIST PLOT OPTIONS
12 = LOG-MODULUS PLOT OPTIONS
13 = ROOT LOCUS PLOT OPTIONS
14 = LIST ROOT LOCUS DATA ON THE PRINTER

APPENDIX B (CONT'D)

```
     ENTER CONTROL DIGIT
1

      ENTER TRANSFER FUNCTION DATA
P2(6)/(1,1)(2,1)
       1     0
     6.00000
       2     1     1
     1.00000   1.00000   2.00000   1.00000

      ENTER TRANSFER FUNCTION DATA
H1(3)/(3,1)
       1     0
     3.00000
       1     1
     3.00000   1.00000

      ENTER TRANSFER FUNCTION DATA

      ILLEGAL TRANSFER FUNCTION DATA

     ENTER CONTROL DIGIT
2
      ENTER TRANSIENT RESPONSE OF INTEREST
      0 = OPEN LOOP
      1 = CLOSED LOOP BEFORE FEEDBACK
      2 = CLOSED LOOP AFTER FEEDBACK
1
      ENTER INPUT VARIABLE
      1 = SETPOINT
      2 = LOAD ONE
      3 = LOAD TWO
      4 = UNIT STEP TO SETPOINT
2
      ENTER INPUT VARIABLE FORCING FUNCTION
      1 = STEP
      2 = PULSE
      3 = SINE
      4 = RAMP
      5 = IMPULSE
1
      ENTER STEP MAGNITUDE
1.0
```

APPENDIX B (CONT'D)

```
     ENTER CONTROL DIGIT
3
      ENTER TIME RESPONSE REPLOT DIGIT
0
      ENTER TOTAL TIME
25
      ENTER NUMBER OF TIME DIVISIONS, MAX.=500
250
      ENTER PLOT DEVICE
1
      ENTER PLOT DIRECTION
3
      ENTER TITLE OF PLOT - 33 CHAR. MAX.

     ENTER CONTROL DIGIT
4
      ENTER DEVICE FOR PERFORMANCE DATA OUTPUT
1

     ENTER CONTROL DIGIT
3
      ENTER TIME RESPONSE REPLOT DIGIT
1
      ENTER PLOT DEVICE
0
      ENTER PLOT DIRECTION
3
      ENTER TITLE OF PLOT - 33 CHAR. MAX.
PROPORTIONAL CONTROL ONLY

     ENTER CONTROL DIGIT
4
      ENTER DEVICE FOR PERFORMANCE CRITERIA
0

     ENTER CONTROL DIGIT
1

      ENTER TRANSFER FUNCTION DATA
C1(0.50)/(1)

      ENTER TRANSFER FUNCTION DATA

      ILLEGAL TRANSFER FUNCTION DATA
```

APPENDIX B (CONT'D)

```
     ENTER CONTROL DIGIT
2
      ENTER TRANSIENT RESPONSE OF INTEREST
1
      ENTER INPUT VARIABLE
2
      ENTER INPUT VARIABLE FORCING FUNCTION
1
      ENTER STEP MAGNITUDE
1.0

     ENTER CONTROL DIGIT
3
      ENTER TIME RESPONSE REPLOT DIGIT
0
      ENTER TOTAL TIME
25
      ENTER NUMBER OF TIME DIVISIONS, MAX.=500
250
      ENTER PLOT DEVICE
1
      ENTER PLOT DIRECTION
3

     ENTER CONTROL DIGIT
4
      ENTER DEVICE FOR PERFORMANCE DATA OUTPUT
2

      PERFORMANCE CRITERIA FOR ABOVE RESPONSE
      STEADY STATE            =      1.1999
      PERCENT OVERSHOOT       =       21.40
      PEAK TIME               =      1.7068
      RISE TIME  0 TO 100     =      1.0126
      RISE TIME 10 TO  90     =      0.6158
      DELAY TIME              =      0.5023
      DECAY RATIO             =      0.0359
      SETTLING TIME           =      2.5100
      ERROR INTEGRALS
           IAE  =       0.88366
           ISE  =       0.55944
           ITAE =       0.86953
```

APPENDIX B (CONT'D)

```
     ENTER CONTROL DIGIT
3
      ENTER TIME RESPONSE REPLOT DIGIT
1
      ENTER PLOT DEVICE
0
      ENTER PLOT DIRECTION
3
      ENTER TITLE OF PLOT - 33 CHAR. MAX.
PROPORTIONAL , KC = 0.50

     ENTER CONTROL DIGIT
1

      ENTER TRANSFER FUNCTION DATA
C1(0.75)/(1)

      ENTER TRANSFER FUNCTION DATA

      ILLEGAL TRANSFER FUNCTION DATA

     ENTER CONTROL DIGIT
2
      ENTER TRANSIENT RESPONSE OF INTEREST
1
      ENTER INPUT VARIABLE
2
      ENTER INPUT VARIABLE FORCING FUNCTION
1
      ENTER STEP MAGNITUDE
1.0

     ENTER CONTROL DIGIT
3
      ENTER TIME RESPONSE REPLOT DIGIT
0
      ENTER TOTAL TIME
25
      ENTER NUMBER OF TIME DIVISIONS, MAX.=500
250
      ENTER PLOT DEVICE
0
      ENTER PLOT DIRECTION
4
      ENTER TITLE OF PLOT - 33 CHAR. MAX.
PROPORTIONAL , KC = 0.75
```

APPENDIX B (CONT'D)

```
     ENTER CONTROL DIGIT
4
      ENTER DEVICE FOR PERFORMANCE DATA OUTPUT
1

     ENTER CONTROL DIGIT
1

      ENTER TRANSFER FUNCTION DATA
C1(0.65)/(1)

      ENTER TRANSFER FUNCTION DATA

      ILLEGAL TRANSFER FUNCTION DATA


     ENTER CONTROL DIGIT
2
      ENTER TRANSIENT RESPONSE OF INTEREST
1
      ENTER INPUT VARIABLE
2
      ENTER INPUT VARIABLE FORCING FUNCTION
1
      ENTER STEP MAGNITUDE
1.0

     ENTER CONTROL DIGIT
3
      ENTER TIME RESPONSE REPLOT DIGIT
0
      ENTER TOTAL TIME
25
      ENTER NUMBER OF TIME DIVISIONS, MAX.=500
500
      ENTER PLOT DEVICE
0
      ENTER PLOT DIRECTION
3
      ENTER TITLE OF PLOT - 33 CHAR. MAX.
PROPORTIONAL , KC = 0.65

     ENTER CONTROL DIGIT
4
      ENTER DEVICE FOR PERFORMANCE DATA OUTPUT
0
```

APPENDIX B (CONT'D)

```
     ENTER CONTROL DIGIT
7
      ENTER LIST DEVICE 0=TYPEWRITER,1=PRINTER
0
      ENTER NAMES OF TRANSFER FUNCTIONS TO BE LISTED
      I.E. P1C1H1H2 ...  OR 99 TO LIST THEM ALL
TM

           *** OVERALL TIME  ***

           NUMERATOR
          36.00000    18.00000     6.00000
           DENOMINATOR
           0.00000    17.69999    11.00000     6.00000     1.00000

     ENTER CONTROL DIGIT
1

      ENTER TRANSFER FUNCTION DATA
C1(1)/(1)

      ENTER TRANSFER FUNCTION DATA

      ILLEGAL TRANSFER FUNCTION DATA

     ENTER CONTROL DIGIT
5
      ENTER FREQUENCY RESPONSE REPLOT DIGIT
0
      ENTER FREQUENCY RESPONSE TYPE
1
      ENTER INITIAL AND TERMINAL FREQUENCIES
0.01 100
      ENTER NUMBER OF POINTS - 100 MAX.
25
      ENTER PLOT DEVICE
1
      ENTER TITLE OF THE BODE PLOT, 36 CHAR. MAX
```

APPENDIX B (CONT'D)

```
     ENTER CONTROL DIGIT
5
      ENTER FREQUENCY RESPONSE REPLOT DIGIT
0
      ENTER FREQUENCY RESPONSE TYPE
1
      ENTER INITIAL AND TERMINAL FREQUENCIES
0.1 100.
      ENTER NUMBER OF POINTS - 100 MAX.
100
      ENTER PLOT DEVICE
0
      ENTER TITLE OF THE BODE PLOT, 36 CHAR. MAX
PROPORTIONAL CONTROL

     ENTER CONTROL DIGIT
5
      ENTER FREQUENCY RESPONSE REPLOT DIGIT
1
      ENTER FREQUENCY RESPONSE TYPE
2
      ENTER TYPE OF SCALE
0
      ENTER PLOT DEVICE
1
      ENTER TITLE OF THE NYQUIST PLOT, 36 CHAR.


     ENTER CONTROL DIGIT
5
      ENTER FREQUENCY RESPONSE REPLOT DIGIT
1
      ENTER FREQUENCY RESPONSE TYPE
2
      ENTER TYPE OF SCALE
1
      ENTER REAL INITIAL AND TERMINAL VALUES
-1.0 3.0
      ENTER IMAGINARY INITIAL AND TERMINAL VALUES
-2.4 0.8
      ENTER PLOT DEVICE
0
      ENTER TITLE OF THE NYQUIST PLOT, 36 CHAR.
PROPORTIONAL CONTROL
```

APPENDIX B (CONT'D)

```
       ENTER CONTROL DIGIT
5
       ENTER FREQUENCY RESPONSE REPLOT DIGIT
1
       ENTER FREQUENCY RESPONSE TYPE
3
       ENTER PLOT DEVICE
0
       ENTER TITLE OF LOG-MODULUS PLOT, 36 CHAR.
PROPORTIONAL CONTROL

      ENTER CONTROL DIGIT
6
       ENTER ROOT LOCUS REPLOT DIGIT
0
       ENTER INITIAL AND TERMINAL FREQUENCIES
-10.0 5.0
       ENTER PLOT DEVICE
1
       ENTER ROOT LOCUS TITLE - 32 CHAR. MAX.


      ENTER CONTROL DIGIT
6
       ENTER ROOT LOCUS REPLOT DIGIT
0
       ENTER INITIAL AND TERMINAL FREQUENCIES
-8 2
       ENTER PLOT DEVICE
0
       ENTER ROOT LOCUS TITLE - 32 CHAR. MAX.
PROPORTIONAL CONTROL

      ENTER CONTROL DIGIT
1

       ENTER TRANSFER FUNCTION DATA
C1(0.65)/(1)

       ENTER TRANSFER FUNCTION DATA
C2(1,0.5)/(0,0.5)

       ENTER TRANSFER FUNCTION DATA

       ILLEGAL TRANSFER FUNCTION DATA
```

APPENDIX B (CONT'D)

```
      ENTER CONTROL DIGIT
2
       ENTER TRANSIENT RESPONSE OF INTEREST
1
       ENTER INPUT VARIABLE
2
       ENTER INPUT VARIABLE FORCING FUNCTION
1
       ENTER STEP MAGNITUDE
1.0

      ENTER CONTROL DIGIT
3
       ENTER TIME RESPONSE REPLOT DIGIT
0
       ENTER TOTAL TIME
25
       ENTER NUMBER OF TIME DIVISIONS, MAX.=500
250
       ENTER PLOT DEVICE
1
       ENTER PLOT DIRECTION
3
       ENTER TITLE OF PLOT - 33 CHAR. MAX.

      ENTER CONTROL DIGIT
3
       ENTER TIME RESPONSE REPLOT DIGIT
1
       ENTER PLOT DEVICE
0
       ENTER PLOT DIRECTION
3
       ENTER TITLE OF PLOT - 33 CHAR. MAX.
PI CONTROL, KC =0.65, TI =0.5

      ENTER CONTROL DIGIT
4
       ENTER DEVICE FOR PERFORMANCE CRITERIA
1
```

APPENDIX B (CONT'D)

```
     ENTER CONTROL DIGIT
1

      ENTER TRANSFER FUNCTION DATA
C2(1,0.75)/(0,0.75)

      ENTER TRANSFER FUNCTION DATA

      ILLEGAL TRANSFER FUNCTION DATA

     ENTER CONTROL DIGIT
2
      ENTER TRANSIENT RESPONSE OF INTEREST
1
      ENTER INPUT VARIABLE
2
      ENTER INPUT VARIABLE FORCING FUNCTION
1
      ENTER STEP MAGNITUDE
1.0

     ENTER CONTROL DIGIT
3
      ENTER TIME RESPONSE REPLOT DIGIT
0
      ENTER TOTAL TIME
25
      ENTER NUMBER OF TIME DIVISIONS, MAX.=500
250
      ENTER PLOT DEVICE
0
      ENTER PLOT DIRECTION
4
      ENTER TITLE OF PLOT - 33 CHAR. MAX.
PI CONTROL, KC =0.65, TI =0.75

     ENTER CONTROL DIGIT
4
      ENTER DEVICE FOR PERFORMANCE CRITERIA
1
```

APPENDIX B (CONT'D)

```
      ENTER CONTROL DIGIT
1

       ENTER TRANSFER FUNCTION DATA
C2(1,1.1)/(0,1.1)

       ENTER TRANSFER FUNCTION DATA

       ILLEGAL TRANSFER FUNCTION DATA

      ENTER CONTROL DIGIT
2
       ENTER TRANSIENT RESPONSE OF INTEREST
1
       ENTER INPUT VARIABLE
2
       ENTER INPUT VARIABLE FORCING FUNCTION
1
       ENTER STEP MAGNITUDE
1.0

      ENTER CONTROL DIGIT
3
       ENTER TIME RESPONSE REPLOT DIGIT
0
       ENTER TOTAL TIME
25
       ENTER NUMBER OF TIME DIVISIONS, MAX.=500
500
       ENTER PLOT DEVICE
0
       ENTER PLOT DIRECTION
3
       ENTER TITLE OF PLOT - 33 CHAR. MAX.
PI CONTROL, KC =0.65, TI =1.10

      ENTER CONTROL DIGIT
4
       ENTER DEVICE FOR PERFORMANCE CRITERIA
0
```

APPENDIX B (CONT'D)

```
     ENTER CONTROL DIGIT
7
      ENTER LIST DEVICE 0=TYPEWRITER,1=PRINTER
0
      ENTER NAMES OF TRANSFER FUNCTIONS TO BE LISTED
      I.E. P1C1H1H2 ...  OR 99 TO LIST THEM ALL
C1P2H1


           *** CONTROL ONE   ***

           NUMERATOR
           0.65000
           DENOMINATOR
           1.00000

           *** PROCESS TWO   ***

           NUMERATOR
           6.00000
           DENOMINATOR
           2.00000      3.00000      1.00000

           *** FEEDBACK ONE  ***

           NUMERATOR
           3.00000
           DENOMINATOR
           3.0000      1.00000

     ENTER CONTROL DIGIT
7
      ENTER LIST DEVICE 0=TYPEWRITER,1=PRINTER
1

      ENTER TITLE FOR TRANSFER FUNCTION LISTING
SAMPLE LISTING TO PRINTER
      ENTER NAMES OF TRANSFER FUNCTIONS TO BE LISTED
      I.E. P1C1H1H2 ...  OR 99 TO LIST THEM ALL
99
```

APPENDIX B (CONT'D)

```
     ENTER CONTROL DIGIT
5
      ENTER FREQUENCY RESPONSE REPLOT DIGIT
0
      ENTER FREQUENCY RESPONSE TYPE
1
      ENTER INITIAL AND TERMINAL FREQUENCIES
0.01 100.
      ENTER NUMBER OF POINTS - 100 MAX.
25
      ENTER PLOT DEVICE
1
      ENTER TITLE OF THE BODE PLOT, 36 CHAR. MAX


     ENTER CONTROL DIGIT
5
      ENTER FREQUENCY RESPONSE REPLOT DIGIT
0
      ENTER FREQUENCY RESPONSE TYPE
1
      ENTER INITIAL AND TERMINAL FREQUENCIES
0.01 1000.0
      ENTER NUMBER OF POINTS - 100 MAX.
100
      ENTER PLOT DEVICE
0
      ENTER TITLE OF THE BODE PLOT, 36 CHAR. MAX
PI CONTROL , KC 0.65 , TI = 1.10

     ENTER CONTROL DIGIT
5
      ENTER FREQUENCY RESPONSE REPLOT DIGIT
1
      ENTER FREQUENCY RESPONSE TYPE
2
      ENTER TYPE OF SCALE
0
      ENTER PLOT DEVICE
1
      ENTER TITLE OF THE NYQUIST PLOT, 36 CHAR.
```

APPENDIX B (CONT'D)

```
      ENTER CONTROL DIGIT
5
       ENTER FREQUENCY RESPONSE REPLOT DIGIT
1
       ENTER FREQUENCY RESPONSE TYPE
2
       ENTER TYPE OF SCALE
1
       ENTER REAL INITIAL AND TERMINAL VALUES
-1 0
       ENTER IMAGINARY INITIAL AND TERMINAL VALUES
-1.4 0.2
       ENTER PLOT DEVICE
0
       ENTER TITLE OF THE NYQUIST PLOT, 36 CHAR.
PI CONTROL , KC = 0.65 , TI = 1.10

      ENTER CONTROL DIGIT
5
       ENTER FREQUENCY RESPONSE REPLOT DIGIT
1
       ENTER FREQUENCY RESPONSE TYPE
3
       ENTER PLOT DEVICE
0
       ENTER TITLE OF LOG-MODULUS PLOT, 36 CHAR.
PI CONTROL , KC = 0.65 , TI = 1.10

      ENTER CONTROL DIGIT
6
       ENTER ROOT LOCUS REPLOT DIGIT
0
       ENTER INITIAL AND TERMINAL FREQUENCIES
-10 5
       ENTER PLOT DEVICE
1
       ENTER ROOT LOCUS TITLE - 32 CHAR. MAX.


      ENTER CONTROL DIGIT
6
       ENTER ROOT LOCUS REPLOT DIGIT
0
       ENTER INITIAL AND TERMINAL FREQUENCIES
-5 0
       ENTER PLOT DEVICE
0
       ENTER ROOT LOCUS TITLE - 32 CHAR. MAX.
PI CONTROL , KC = 0.65 , TI = 1.10
```

APPENDIX B (CONT'D)

```
      ENTER CONTROL DIGIT
1

      ENTER TRANSFER FUNCTION DATA
C1(0.65)(1,1.1)/(0,1.1)

      ENTER TRANSFER FUNCTION DATA
C2(1,1.5)/(1)

      ENTER TRANSFER FUNCTION DATA

      ILLEGAL TRANSFER FUNCTION DATA

     ENTER CONTROL DIGIT
2
      ENTER TRANSIENT RESPONSE OF INTEREST
1
      ENTER INPUT VARIABLE
2
      ENTER INPUT VARIABLE FORCING FUNCTION
1
      ENTER STEP MAGNITUDE
1.0

      ENTER CONTROL DIGIT
3
      ENTER TIME RESPONSE REPLOT DIGIT
0
      ENTER TOTAL TIME
25
      ENTER NUMBER OF TIME DIVISIONS, MAX.=500
250
      ENTER PLOT DEVICE
1
      ENTER PLOT DIRECTION
3
      ENTER TITLE OF PLOT - 33 CHAR. MAX.

      ENTER CONTROL DIGIT
4
      ENTER DEVICE FOR PERFORMANCE CRITERIA
1
```

APPENDIX B (CONT'D)

```
      ENTER CONTROL DIGIT
1

       ENTER TRANSFER FUNCTION DATA
C2(1,0.5)/(1)

       ENTER TRANSFER FUNCTION DATA

       ILLEGAL TRANSFER FUNCTION DATA

      ENTER CONTROL DIGIT
2
       ENTER TRANSIENT RESPONSE OF INTEREST
1
       ENTER INPUT VARIABLE
2
       ENTER INPUT VARIABLE FORCING FUNCTION
1
       ENTER STEP MAGNITUDE
1.0

      ENTER CONTROL DIGIT
3
       ENTER TIME RESPONSE REPLOT DIGIT
0
       ENTER TOTAL TIME
25
       ENTER NUMBER OF TIME DIVISIONS, MAX.=500
250
       ENTER PLOT DEVICE
0
       ENTER PLOT DIRECTION
3
       ENTER TITLE OF PLOT - 33 CHAR. MAX.
PID CONTROL,KC=.65,TI=1.1,TD=.5

      ENTER CONTROL DIGIT
4
       ENTER DEVICE FOR PERFORMANCE CRITERIA
1
```

APPENDIX B (CONT'D)

```
      ENTER CONTROL DIGIT
1

       ENTER TRANSFER FUNCTION DATA
C2(1,0.25)/(1)

       ENTER TRANSFER FUNCTION DATA

       ILLEGAL TRANSFER FUNCTION DATA

      ENTER CONTROL DIGIT
2
       ENTER TRANSIENT RESPONSE OF INTEREST
1
       ENTER INPUT VARIABLE
2
       ENTER INPUT VARIABLE FORCING FUNCTION
1
       ENTER STEP MAGNITUDE
1.0

      ENTER CONTROL DIGIT
3
       ENTER TIME RESPONSE REPLOT DIGIT
0
       ENTER TOTAL TIME
25
       ENTER NUMBER OF TIME DIVISIONS, MAX.=500
500
       ENTER PLOT DEVICE
0
       ENTER PLOT DIRECTION
3
       ENTER TITLE OF PLOT - 33 CHAR. MAX.
PID CONTROL,KC=.65,TI=1.1,TD=.25

      ENTER CONTROL DIGIT
4
       ENTER DEVICE FOR PERFORMANCE DATA OUTPUT
1
```

APPENDIX B (CONT'D)

```
      ENTER CONTROL DIGIT
1
      ENTER TRANSFER FUNCTION DATA
C2(1,0.01)/(1)
      ENTER TRANSFER FUNCTION DATA

      ILLEGAL TRANSFER FUNCTION DATA

     ENTER CONTROL DIGIT
2
      ENTER TRANSIENT RESPONSE OF INTEREST
1
      ENTER INPUT VARIABLE
2
      ENTER INPUT VARIABLE FORCING FUNCTION
1
      ENTER STEP MAGNITUDE
1.0

     ENTER CONTROL DIGIT
3
      ENTER TIME RESPONSE REPLOT DIGIT
0
      ENTER TOTAL TIME
25
      ENTER NUMBER OF TIME DIVISIONS, MAX.=500
250
      ENTER PLOT DEVICE
1
      ENTER PLOT DIRECTION
3
      ENTER TITLE OF PLOT - 33 CHAR. MAX.


     ENTER CONTROL DIGIT
4
      ENTER DEVICE FOR PERFORMANCE DATA OUTPUT
1

     ENTER CONTROL DIGIT
3
      ENTER TIME RESPONSE REPLOT DIGIT
1
      ENTER PLOT DEVICE
0
      ENTER PLOT DIRECTION
4
      ENTER TITLE OF PLOT - 33 CHAR. MAX.
PID CONTROL,KC=.65,TI=1.1,TD=.01
```

APPENDIX B (CONT'D)

```
     ENTER CONTROL DIGIT
4
      ENTER DEVICE FOR PERFORMANCE CRITERIA
0

     ENTER CONTROL DIGIT
5
      ENTER FREQUENCY RESPONSE REPLOT DIGIT
0
      ENTER FREQUENCY RESPONSE TYPE
1
      ENTER INITIAL AND TERMINAL FREQUENCIES
0.001 1000.0
      ENTER NUMBER OF POINTS - 100 MAX.
50
      ENTER PLOT DEVICE
1
      ENTER TITLE OF THE BODE PLOT, 36 CHAR. MAX


     ENTER CONTROL DIGIT
5
      ENTER FREQUENCY RESPONSE REPLOT DIGIT
0
      ENTER FREQUENCY RESPONSE TYPE
1
      ENTER INITIAL AND TERMINAL FREQUENCIES
.01 10000
      ENTER NUMBER OF POINTS - 100 MAX.
100
      ENTER PLOT DEVICE
0
      ENTER TITLE OF THE BODE PLOT, 36 CHAR. MAX
PID CONTROL,KC=.65,TI=1.1,TD=.01

     ENTER CONTROL DIGIT
5
      ENTER FREQUENCY RESPONSE REPLOT DIGIT
1
      ENTER FREQUENCY RESPONSE TYPE
2
      ENTER TYPE OF SCALE
0
      ENTER PLOT DEVICE
1
      ENTER TITLE OF THE NYQUIST PLOT, 36 CHAR.
```

APPENDIX B (CONT'D)

```
     ENTER CONTROL DIGIT
5
      ENTER FREQUENCY RESPONSE REPLOT DIGIT
1
      ENTER FREQUENCY RESPONSE TYPE
2
      ENTER TYPE OF SCALE
1
      ENTER REAL INITIAL AND TERMINAL VALUES
-1 0
      ENTER IMAGINARY INITIAL AND TERMINAL VALUES
-0.9 0.3
      ENTER PLOT DEVICE
0
      ENTER TITLE OF THE NYQUIST PLOT, 36 CHAR.
PID CONTROL,KC=.65,TI=1.1,TD=.01

     ENTER CONTROL DIGIT
5
      ENTER FREQUENCY RESPONSE REPLOT DIGIT

      ENTER FREQUENCY RESPONSE TYPE
3
      ENTER PLOT DEVICE
0
      ENTER TITLE OF LOG-MODULUS PLOT, 36 CHAR.
PID CONTROL,KC=.65,TI=1.1,TD=.01

     ENTER CONTROL DIGIT
6
      ENTER ROOT LOCUS REPLOT DIGIT
0
      ENTER INITIAL AND TERMINAL FREQUENCIES
-5 0
      ENTER PLOT DEVICE
1
      ENTER ROOT LOCUS TITLE - 32 CHAR. MAX.


     ENTER CONTROL DIGIT
6
      ENTER ROOT LOCUS REPLOT DIGIT
1
      ENTER PLOT DEVICE
0
      ENTER ROOT LOCUS TITLE - 32 CHAR. MAX.
PID CONTROL,KC=.65,TI=1.1,TD=.01
```

TRANSIENT RESPONSE DATA

*** PROPORTIONAL , KC = 0.65 ***

START =0.0000 TOTAL TIME = 25.00000 STEP SIZE = 0.10040

0.0000	0.0273	0.2019	0.3248	0.4584
0.5951	0.7287	0.8542	0.9679	1.0670
1.1498	1.2157	1.2646	1.2972	1.3145
1.3183	1.3103	1.2926	1.2672	1.2362
1.2016	1.1653	1.1287	1.0934	1.0604
1.0306	1.0047	0.9831	0.9659	0.9531
0.9446	0.9400	0.9390	0.9410	0.9455
0.9521	0.9600	0.9690	0.9784	0.9878
0.9970	1.0055	1.0132	1.0199	1.0256
1.0300	1.0333	1.0356	1.0368	1.0371
1.0365	1.0354	1.0337	1.0316	1.0293
1.0269	1.0245	1.0221	1.0199	1.0179
1.0161	1.0147	1.0135	1.0127	1.0121
1.0118	1.0117	1.0118	1.0121	1.0125
1.0131	1.0137	1.0143	1.0149	1.0155
1.0161	1.0166	1.0171	1.0175	1.0178
1.0180	1.0181	1.0182	1.0182	1.0182
1.0181	1.0180	1.0179	1.0177	1.0176
1.0174	1.0173	1.0171	1.0170	1.0169
1.0168	1.0167	1.0166	1.0166	1.0166
1.0166	1.0166	1.0166	1.0166	1.0166
1.0167	1.0167	1.0168	1.0168	1.0168
1.0169	1.0169	1.0169	1.0170	1.0170
1.0170	1.0170	1.0170	1.0170	1.0170
1.0170	1.0170	1.0170	1.0169	1.0169
1.0169	1.0169	1.0169	1.0169	1.0169
1.0169	1.0169	1.0169	1.0169	1.0169
1.0169	1.0169	1.0169	1.0169	1.0169
1.0169	1.0169	1.0169	1.0169	1.0169
1.0169	1.0169	1.0169	1.0169	1.0169
1.0169	1.0169	1.0169	1.0169	1.0169
1.0169	1.0169	1.0169	1.0169	1.0169
1.0169	1.0169	1.0169	1.0169	1.0169
1.0169	1.0169	1.0169	1.0169	1.0169
1.0169	1.0169	1.0169	1.0169	1.0169
1.0169	1.0169	1.0169	1.0169	1.0169
1.0169	1.0169	1.0169	1.0169	1.0169
1.0169	1.0169	1.0169	1.0169	1.0169
1.0169	1.0169	1.0169	1.0169	1.0169
1.0169	1.0169	1.0169	1.0169	1.0169
1.0169	1.0169	1.0169	1.0169	1.0169
1.0169	1.0169	1.0169	1.0169	1.0169
1.0169	1.0169	1.0169	1.0169	1.0169
1.0169	1.0169	1.0169	1.0169	1.0169
1.0169	1.0169	1.0169	1.0169	1.0169
1.0169	1.0169	1.0169	1.0169	1.0169
1.0169	1.0169	1.0169	1.0169	1.0169
1.0169	1.0169	1.0169	1.0169	1.0169
1.0169	1.0169	1.0169	1.0169	1.0169
1.0169	1.0169	1.0169	1.0169	1.0169
1.0169	1.0169	1.0169	1.0169	1.0169

```
***  SAMPLE LISTING TO PRINTER                         ***


*** CONTROL ONE    ***

NUMERATOR
0.65000
DENOMINATOR
1.00000

*** CONTROL TWO    ***

NUMERATOR
1.00000      1.10000
DENOMINATOR
0.00000      1.10000

*** PROCESS ONE    ***

NUMERATOR
1.00000
DENOMINATOR
1.00000

*** PROCESS TWO    ***

NUMERATOR
6.00000
DENOMINATOR
2.00000      3.00000      1.00000

*** PROCESS 3      ***

NUMERATOR
1.00000
DENOMINATOR
1.00000

*** FEEDBACK ONE   ***

NUMERATOR
3.00000
DENOMINATOR
3.00000      1.00000
```

```
***  SAMPLE LISTING TO PRINTER                           ***


*** FEEDBACK TWO   ***

NUMERATOR
0.00000
DENOMINATOR
1.00000

*** LOAD ONE        ***

NUMERATOR
1.00000
DENOMINATOR
1.00000

*** LOAD TWO        ***

NUMERATOR
1.00000
DENOMINATOR
1.00000

*** OPEN LOOP       ***

NUMERATOR
11.69999     12.86999
DENOMINATOR
0.00000      6.60000     12.09999      6.60000      1.10000

*** CLOSED LOOP    ***

NUMERATOR
11.69999     16.77000     4.28999
DENOMINATOR
11.69999     19.46999    12.09999      6.60000      1.10000

*** FORWARD PATH   ***

NUMERATOR
3.89999      4.28999
DENOMINATOR
0.00000      2.20000      3.30000      1.10000
```

```
***  SAMPLE LISTING TO PRINTER                         ***


*** SETPOINT FF    ***

NUMERATOR
1.00000
DENOMINATOR
1.00000

*** LOAD 1 FPATH   ***

NUMERATOR
6.00000
DENOMINATOR
2.00000      3.00000      1.00000

*** LOAD 1 * FF    ***

NUMERATOR
-0.00002     19.80000     6.60000
DENOMINATOR
11.69999     19.46999     12.09999     6.60000      1.10000

*** LOAD 2 FPATH   ***

NUMERATOR
1.00000
DENOMINATOR
1.00000

*** LOAD 2 * FF    ***

NUMERATOR
1.00000
DENOMINATOR
1.00000

*** OVERALL TIME   ***

NUMERATOR
-0.00001     18.00000     6.00000
DENOMINATOR
0.00000      10.63636     17.69999     10.99999     6.00000
1.00000
```

FREQUENCY RESPONSE DATA

*** PI CONTROL , KC 0.65 , TI = 1.10 *** PAGE 1 OF 2

FREQUENCY (RADIANS/SECOND)	MAGNITUDE (DECIBELS)	PHASE ANGLE (DEGREES)
0.01001000	44.96410548	-90.42067527
0.01124323	43.95493960	-90.47250223
0.01262856	42.94566613	-90.53066539
0.01418473	41.93629264	-90.59608745
0.01593281	40.92682695	-90.66952180
0.01789648	39.91728430	-90.75202989
0.02010231	38.90766477	-90.84474945
0.02258017	37.89796835	-90.94888019
0.02536361	36.88821041	-91.06582164
0.02849031	35.87836790	-91.19723463
0.03200262	34.86846387	-91.34481096
0.03594807	33.85848295	-91.51064205
0.04038009	32.84843277	-91.69692707
0.04535869	31.83827888	-91.90621900
0.05095126	30.82803651	-92.14131689
0.05723353	29.81767496	-92.40541934
0.06429057	28.80717116	-92.70223283
0.07221791	27.79650202	-93.03566360
0.08112285	26.78562524	-93.41029453
0.09112599	25.77449092	-93.83132362
0.10236277	24.76303359	-94.30441045
0.11498532	23.75117260	-94.83613729
0.12916453	22.73880025	-95.43385505
0.14509234	21.72577428	-96.10585308
0.16298444	20.71191400	-96.86140489
0.18308309	19.69697338	-97.71113729
0.20566039	18.68064093	-98.66701507
0.23102197	17.66250532	-99.74257183
0.25951127	16.64200136	-100.95314025
0.29151393	15.61839471	-102.31588315
0.32746334	14.59069918	-103.85030126
0.36784615	13.55758832	-105.57812476
0.41320908	12.51730896	-107.52337574
0.46416641	11.46751581	-109.71236753
0.52140807	10.40516576	-112.17324352
0.58570873	9.32631602	-114.93520832
0.65793951	8.22597999	-118.02751278
0.73907782	7.09801860	-121.47768497
0.83022242	5.93505954	-125.30934715
0.93260745	4.72853195	-129.53998494
1.04761895	3.46884851	-134.17821073
1.17681451	2.14563612	-139.22145628
1.32194218	0.74819129	-144.65406584
1.48496804	-0.73407042	-150.44614267
1.66809850	-2.31102361	-156.55305695
1.87381365	-3.99125082	-162.91649198
2.10489876	-5.78150278	-169.46628904
2.36448176	-7.68621200	-176.12349247
2.65607698	-9.70721769	-182.80437946
2.98363392	-11.84361349	-189.42488884

*** PI CONTROL , KC 0.65 , TI = 1.10 ***

FREQUENCY (RADIANS/SECOND)	MAGNITUDE (DECIBELS)	PHASE ANGLE (DEGREES)
3.35158508	-14.09186221	-195.90501928
3.76491419	-16.44610831	-202.17307424
4.22921623	-18.89860662	-208.16864514
4.75078047	-21.44031429	-213.84451866
5.33666317	-24.06146955	-219.16753768
5.99479902	-26.75214916	-224.11786389
6.73409912	-29.50271433	-228.68759958
7.56457325	-32.30414146	-232.88010549
8.49746629	-35.14826053	-236.70497012
9.54540628	-38.02782040	-240.17879509
10.72257740	-40.93657720	-243.32178854
12.04492570	-43.86920207	-246.15664958
13.53035031	-46.82121974	-248.70706152
15.19896689	-49.78892397	-250.99667787
17.07335749	-52.76924663	-253.04878163
19.17891082	-55.75972706	-254.88546603
21.54412597	-58.75831222	-256.52761983
24.20102956	-61.76338726	-257.99446296
27.18558913	-64.77365267	-259.30388689
30.53822430	-67.78803193	-260.47200965
34.30431061	-70.80567157	-261.51371717
38.53484857	-73.82595634	-262.44235897
43.28711694	-76.84828662	-263.26986685
48.62544590	-79.87227773	-264.00725793
54.62211638	-82.89756071	-264.66412305
61.35833680	-85.92388951	-265.24913787
68.92527353	-88.95104861	-265.77017641
77.42541193	-91.97883844	-266.23418951
86.97380948	-95.00716638	-266.64732933
97.69974148	-98.03589427	-267.01525497
109.74844658	-101.06494510	-267.34282636
123.28306448	-104.09425735	-267.63441133
138.48680448	-107.12376952	-267.89406919
155.56551384	-110.15345084	-268.12518405
174.75043034	-113.18327057	-268.33101558
196.30136537	-116.21318268	-268.51420927
220.50999593	-119.24317157	-268.67734909
247.70410895	-122.27322196	-268.82252645
278.25190639	-125.30331861	-268.95183229
312.56705141	-128.33347654	-269.06686687
351.11404943	-131.36365008	-269.16935205
394.41480016	-134.39388489	-269.26051855
443.05542516	-137.42408919	-269.34171915
497.69481945	-140.45432424	-269.41400003
559.07246398	-143.48455929	-269.47828435
628.01943778	-146.51485562	-269.53561687
705.46911811	-149.54512143	-269.58655214
792.47034835	-152.57538723	-269.63201189
890.20090866	-155.60565304	-270.00006103
999.98388099	-158.63594937	-270.00006103

*** PI CONTROL , KC = 0.65 , TI = 1. *** PAGE 1 OF 2

REAL PART	IMAG. PART	GAIN
-5.0000009	5.5196754	-23.2936376
-4.9000013	5.3399658	-21.1616518
-4.8000007	5.1598140	-19.1610904
-4.7000001	4.9791721	-17.2878436
-4.5999996	4.7979851	-15.5378129
-4.4999999	4.6161897	-13.9068901
-4.3999994	4.4337119	-12.3909921
-4.2999988	4.2504623	-10.9859936
-4.1999982	4.0663399	-9.6878114
-4.0999976	3.8812202	-8.4923431
-3.9999971	3.6949563	-7.3954895
-3.8999965	3.5073684	-6.3931461
-3.7999959	3.3182329	-5.4812223
-3.6999958	3.1272746	-4.6556232
-3.5999952	2.9341430	-3.9122501
-3.4999951	2.7383840	-3.2470131
-3.3999946	2.5394013	-2.6558227
-3.2999940	2.3363882	-2.1345983
-3.1999939	2.1282161	-1.6792592
-3.0999933	1.9132404	-1.2857365
-2.9999932	1.6889297	-0.9499363
-2.8999926	1.4510798	-0.6679277
-2.7999921	1.1918419	-0.4357587
-2.6999920	0.9933097	-0.2494122
-2.5999914	0.4898504	-0.1052350
-1.1999879	0.1898473	0.1226379
-1.0999878	0.3284905	0.1638061
-0.9999876	0.4264140	0.2020199
-0.8999876	0.5379951	0.2419348
-0.7999876	0.7131983	0.3036091
-0.6999875	0.9549648	0.4162622
-0.5999875	1.2098698	0.5857809
-0.4999874	1.4538648	0.8081114
-0.3999874	1.6852585	1.0831142
-0.2999874	1.9064424	1.4131400
-0.1999873	2.1197596	1.8015040
-0.0999873	2.3269932	2.2519189
-5.0000009	0.0000000	0.0523793
-4.9000013	0.0000000	2.5071041
-4.8000007	0.0000000	2.2550226
-4.7000001	0.0000000	2.0193572
-4.5999996	0.0000000	1.7996049
-4.4999999	0.0000000	1.5952278
-4.3999994	0.0000000	1.4057525
-4.2999988	0.0000000	1.2306285
-4.1999982	0.0000000	1.0693765
-4.0999976	0.0000000	0.9214622
-3.9999971	0.0000000	0.7863800
-3.8999965	0.0000000	0.6635965
-3.7999959	0.0000000	0.5526250

ROOT LOCUS POINT DATA

*** PI CONTROL , KC = 0.65 , TI = 1. *** PAGE 2 OF 2

REAL PART	IMAG. PART	GAIN
-3.6999958	0.0000000	0.4529252
-3.5999952	0.0000000	0.3639973
-3.4999951	0.0000000	0.2853233
-3.3999946	0.0000000	0.2103734
-3.2999940	0.0000000	0.1566603
-3.1999939	0.0000000	0.1056586
-3.0999933	0.0000000	0.0627854
-2.9999932	0.0000000	0.0271902
-2.8999926	0.0000000	8.9515101
-2.7999921	0.0000000	-0.0206037
-2.6999920	0.0000000	-0.0361332
-2.5999914	0.0000000	-0.0457973
-2.4999913	0.0000000	-0.0503550
-2.3999907	0.0000000	-0.0501974
-2.2999901	0.0000000	-0.0460539
-2.1999900	0.0000000	-0.0384473
-2.0999895	0.0000000	-0.0277005
-1.9999891	0.0000000	-0.0145342
-1.8999890	0.0000000	-1.7597870
-1.7999887	0.0000000	0.0158490
-1.6999886	0.0000000	0.0329788
-1.5999885	0.0000000	0.0500927
-1.4999884	0.0000000	0.0664487
-1.3999883	0.0000000	0.0813204
-1.2999880	0.0000000	0.0935535
-1.1999879	0.0000000	0.1014224
-1.0999878	0.0000000	0.1014730
-0.9999876	0.0000000	0.0841008
-0.8999876	0.0000000	2.6646733
-0.7999876	0.0000000	1.9494193
-0.6999875	0.0000000	0.3308899
-0.5999875	0.0000000	0.2566516
-0.4999874	0.0000000	0.2229814
-0.3999874	0.0000000	0.1958646
-0.2999874	0.0000000	0.1676150
-0.1999873	0.0000000	0.1352539
-0.0999873	0.0000000	0.0971939